

日 本 国 特 許 庁
JAPAN PATENT OFFICE

添付の書類に記載されている事項は下記の出願書類に記載されて
項と同一であることを証明する。

is to certify that the annexed is a true copy of the following application as filed
wit s Office

日 年 月 日
Date of Application:

2002年 7月30日

出 願 番 号
App. tion Number:

特願2002-220943

[ST.1 'C]:

[JP2002-220943]

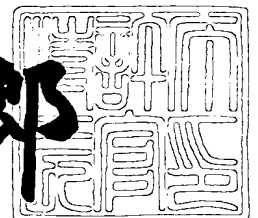
出 願 人
Applic (s):

ヤマハ株式会社

2003年 5月23日

特 許 庁 長 官
Commissioner,
Japan Patent Office

太田 信一郎



出証番号 出証特2003-3038014

【書類名】 特許願

【整理番号】 C30587

【あて先】 特許庁長官殿

【国際特許分類】 H04S 7/00

【発明者】

 【住所又は居所】 静岡県浜松市中沢町 1 0 番 1 号 ヤマハ株式会社内

 【氏名】 青木 孝光

【発明者】

 【住所又は居所】 静岡県浜松市中沢町 1 0 番 1 号 ヤマハ株式会社内

 【氏名】 中山 圭

【特許出願人】

 【識別番号】 000004075

 【氏名又は名称】 ヤマハ株式会社

【代理人】

 【識別番号】 100104798

 【弁理士】

 【氏名又は名称】 山下 智典

【手数料の表示】

 【予納台帳番号】 085513

 【納付金額】 21,000円

【提出物件の目録】

 【物件名】 明細書 1

 【物件名】 図面 1

 【物件名】 要約書 1

【ブルーフの要否】 要

【書類名】 明細書

【発明の名称】 ミキシング方法、双方向カスケード接続デジタルミキサおよびプログラム

【特許請求の範囲】

【請求項 1】 一のデジタルミキサに適用されるミキシング方法であって、
複数の入力信号を加算し入力加算信号を出力する第 1 の加算過程と、
前記入力加算信号をカスケード信号として出力するカスケード出力過程と、
他のデジタルミキサから出力されるカスケード信号を入力するためのカスケード入力過程と、
前記入力加算信号を遅延する遅延過程と、
遅延された前記入力加算信号と、入力されたカスケード信号とを加算しミキシング出力信号として出力する第 2 の加算過程と
を有することを特徴とするミキシング方法。

【請求項 2】 複数のミキシング系列を備えた一のデジタルミキサに適用されるミキシング方法であって、前記各ミキシング系列毎に
複数の入力信号を加算し入力加算信号を出力する第 1 の加算過程と、
前記入力加算信号をカスケード信号として出力するカスケード出力過程と、
他のデジタルミキサから出力されるカスケード信号を入力するためのカスケード入力過程と、
前記入力加算信号を遅延する遅延過程と、
リンクをオンまたはオフ状態に設定するオンオフ過程と、
前記リンクがオン状態に設定された場合は、遅延された前記入力加算信号と入力されたカスケード信号とを加算しミキシング出力信号として出力する一方、前記リンクがオフ状態に設定された場合は遅延された前記入力加算信号をそのままミキシング出力信号として出力する第 2 の加算過程と
を実行することを特徴とするミキシング方法。

【請求項 3】 前記一のデジタルミキサと、前記他のデジタルミキサとが協同して動作可能であるか否かを判定する判定過程をさらに具備し、
前記第 2 の加算過程は、該判定過程において協同して動作可能であると判定さ

れたことを条件として、遅延された前記入力加算信号と、入力されたカスケード信号とを加算し前記ミキシング出力信号として出力するものである

ことを特徴とする請求項 1 または 2 記載のミキシング方法。

【請求項 4】 請求項 1 ないし 3 の何れかに記載のミキシング方法を実行することを特徴とする双方向カスケード接続デジタルミキサ。

【請求項 5】 請求項 1 ないし 3 の何れかに記載のミキシング方法を実行することを特徴とするプログラム。

【発明の詳細な説明】

【 0 0 0 1 】

【発明の属する技術分野】

本発明は、大規模なミキシングシステムに用いて好適なミキシング方法、双方向カスケード接続デジタルミキサおよびプログラムに関する。

【 0 0 0 2 】

【従来の技術】

近年、特に業務用の音響設備においてデジタルミキシングシステムが普及しつつある。このシステムにおいては、マイク等から収集された音声信号が全てデジタル信号に変換され、DSP アレイ等によって構成されたエンジンにおいてミキシング処理が行われる。そして、大型のデジタルミキシングシステムにおいては、オペレータが操作するミキシングコンソールと、上記エンジンとが分離されていることが多い。

【 0 0 0 3 】

例えば、ミキシングコンソールは客席内の中央あるいは客席から分離されたミキシングルームに載置され、エンジンは舞台裏等に載置される。ミキシングコンソールにおいては複数のフェーダ等の操作子が設けられており、これらは全て CPU によって自動的に駆動可能である。例えば、舞台転換が行われた場合には、その時々舞台状況に応じてフェーダ等の操作位置を予め決定しておいた位置に自動的に設定することができる。かかる操作を「シーンリコール」と呼ぶ。

【 0 0 0 4 】

シーンリコールあるいはオペレータのマニュアル操作によってフェーダ等の操

作量が変更されると、その情報はミキシングコンソールからエンジンに通知され、これによってエンジン内におけるアルゴリズムあるいは演算パラメータ等が決定される。また、デジタルミキシングシステムに対して要求される処理能力はコンサート等の規模に応じて様々であるため、複数のコンソールおよび複数のエンジンを組み合わせることにより、処理能力を向上できれば便利である。このため、複数のミキシングシステムをカスケード接続し処理能力を向上させる技術が開 2 0 0 0 - 2 6 1 3 9 1 号公報等に開示されている。

【 0 0 0 5 】

【発明が解決しようとする課題】

ところで、上述した従来のカスケード接続技術においては、最終的なミキシング結果は最下流のミキシングシステム（カスケードマスタ）においてのみ得られるものであった。このため、複数のミキシングシステムをカスケード接続しつつ各々のシステムにおいて独立したミキシング結果を得ることは不可能であった。同様に、カスケード接続されたミキシングシステムにおけるキュー信号が複数段階に渡ってミキシングされている場合に、最終的なキュー信号は最下流のミキシングシステム（カスケードマスタ）においてのみ得られるものであり、各々のシステムにおいて独立して最終的なキュー信号を得ることも困難であった。

この発明は上述した事情に鑑みてなされたものであり、複数のミキシングシステムを用いて処理能力を向上させつつ、各ミキシングシステムにおいて高い独立性を確保できるミキシング方法、双方向カスケード接続デジタルミキサおよびプログラムを提供することを目的としている。

【 0 0 0 6 】

【課題を解決するための手段】

上記課題を解決するため本発明にあっては、下記構成を具備することを特徴とする。なお、括弧内は例示である。

請求項 1 記載のミキシング方法にあっては、一のデジタルミキサに適用されるミキシング方法であって、複数の入力信号を加算し入力加算信号を出力する第 1 の加算過程（ミキシングバス 2 4 4 e）と、前記入力加算信号をカスケード信号として出力するカスケード出力過程（2 4 4 e から加算器 2 6 6 f への信号出力

）と、他のデジタルミキサから出力されるカスケード信号を入力するためのカスケード入力過程（ミキシングバス 2 4 4 f から加算器 2 6 6 e への信号入力）と、前記入力加算信号を遅延する遅延過程（遅延回路 2 6 4 e）と、遅延された前記入力加算信号と、入力されたカスケード信号とを加算しミキシング出力信号として出力する第 2 の加算過程（加算器 2 6 6 e）とを有することを特徴とする。

また、請求項 2 記載のミキシング方法にあっては、複数のミキシング系列（第 1、第 2 キュー信号 CUE 1、CUE 2、ミキシング出力）を備えた一のデジタルミキサに適用されるミキシング方法であって、前記各ミキシング系列毎に複数の入力信号を加算し入力加算信号を出力する第 1 の加算過程と、前記入力加算信号をカスケード信号として出力するカスケード出力過程と、他のデジタルミキサから出力されるカスケード信号を入力するためのカスケード入力過程と、前記入力加算信号を遅延する遅延過程と、リンクをオンまたはオフ状態に設定するオンオフ過程（2 7 4 e、2 7 4 f、2 8 0 e、2 8 0 f）と、前記リンクがオン状態に設定された場合は、遅延された前記入力加算信号と入力されたカスケード信号とを加算しミキシング出力信号として出力する一方、前記リンクがオフ状態に設定された場合は遅延された前記入力加算信号をそのままミキシング出力信号として出力する第 2 の加算過程とを実行することを特徴とする。

さらに、請求項 3 記載の構成にあっては、請求項 1 または 2 記載のミキシング方法において、前記一のデジタルミキサと、前記他のデジタルミキサとが協同して（カスケード接続により）動作可能であるか否かを判定する判定過程（CPU 1 1 8、SP 2 1 2、SP 2 1 4）をさらに具備し、前記第 2 の加算過程は、該判定過程において協同して動作可能であると判定されたことを条件として、遅延された前記入力加算信号と、入力されたカスケード信号とを加算し前記ミキシング出力信号として出力するものであることを特徴とする。

また、請求項 4 記載の双方向カスケード接続デジタルミキサにあっては、請求項 1 ないし 3 の何れかに記載のミキシング方法を実行することを特徴とする。

また、請求項 5 記載のプログラムにあっては、請求項 1 ないし 3 の何れかに記載のミキシング方法を実行することを特徴とする。

【0 0 0 7】

【発明の実施の形態】

1. 実施形態のハードウェア構成

1. 1. コンソール

次に、本発明の一実施形態のデジタルミキシングシステムについて説明する。本実施形態は、一または複数のコンソール 1 0 0 と、一または複数のエンジン 2 0 0 とにによって構成される。まず、コンソール 1 0 0 のハードウェア構成を図 1 (a) を参照し説明する。

【0 0 0 8】

図において 1 0 2 は表示器であり、コンソール 1 0 0 のオペレータに対して各種の情報を表示する。1 0 4 は電動フェーダ部であり、「4 8」個の電動フェーダによって構成されている。これら電動フェーダは、オペレータによって操作されるとともに、コンソール 1 0 0 内部に記憶されたシーンデータ等に基づいて、必要な場合には自動的に駆動される。

【0 0 0 9】

1 1 4 は操作子群であり、音声信号の音質等を調節する各種の操作子等によって構成されている。これら操作子もオペレータによって操作されるとともに、コンソール 1 0 0 内部に記憶されたデータ等に基づいて、必要な場合には自動的に駆動される。さらに、操作子群 1 1 4 には、文字入力用のキーボードおよびマウス等も含まれており、表示器 1 0 2 上にはこのマウスに対応するマウスカーソルが表示される。1 0 6 はデュアル I / O 部であり、デュアルコンソールシステム（詳細は後述する）を構成する場合に、ここを介して他のコンソールが接続され、当該他のコンソールとの間でデジタル音声信号、制御信号等の入出力を行う。

【0 0 1 0】

1 1 0 はデータ I / O 部であり、エンジン 2 0 0 との間でデジタル音声信号の入出力を行う。これらデジタル音声信号は、例えばオペレータの声等であるトークバック信号、エンジン 2 0 0 側の作業員の声等であるコムイン信号およびエンジン 2 0 0 のモニタ信号などである。1 0 8 は波形 I / O 部であり、エンジン 2 0 0 から供給されたデジタル音声信号をアナログ信号に変換するとともに、トークバックマイク（図示せず）を介して入力されたトークバック信号（アナログ）

をデジタル信号に変換しデータ I / O 部 1 1 0 に供給する。

【 0 0 1 1 】

1 1 2 は通信 I / O 部であり、エンジン 2 0 0 との間で各種の制御信号を入出力する。コンソール 1 0 0 側から送信される制御信号には、電動フェーダ部 1 0 4 および操作子群 1 1 4 等の操作情報が含まれている。これら操作情報によって、エンジン 2 0 0 側のアルゴリズムに用いられるパラメータが設定される。1 1 6 はその他 I / O 部であり、オペレータ側に設けられた各種外部機器が接続される。1 1 8 は CPU であり、フラッシュメモリ 1 2 0 に記憶されたプログラムに基づいて、バス 1 2 4 を介して各部を制御する。

【 0 0 1 2 】

1 2 2 は RAM であり、CPU 1 1 8 のワークメモリとして用いられる。ここで、RAM 1 2 2 に格納されるデータについて詳述しておく。RAM 1 2 2 においては、カレント領域 1 2 2 a、シーン領域 1 2 2 b およびライブラリ領域 1 2 2 c が確保される。このカレント領域 1 2 2 a には、ミキシングコンソールの現在の設定状態、例えば各入力チャンネルの減衰量、周波数特性の設定量等、出力チャンネルの減衰量、各種エフェクトの設定内容等が記憶される。これらデータを「カレント操作データ」と呼ぶ。このカレント操作データが更新されると、これによってエンジン 2 0 0 による信号処理の内容等も決定される。

【 0 0 1 3 】

また、シーン領域 1 2 2 b にはカレント操作データと同一構造のデータを複数セット（最大約「1 0 0 0」セット）記憶することができる。例えば、ある時点のカレント領域 1 2 2 a の内容（シーン）をシーン領域 1 2 2 b 内に格納しておくことにより、その時点の設定状態をワンタッチで再現（リコール）することができる。これらのデータを「シーンデータ」と呼ぶ。また、ライブラリ領域 1 2 2 c には、エンジン 2 0 0 におけるユニット構成を規定したユニットライブラリ、入出力パッチ（詳細は後述する）における接続関係を規定したパッチライブラリ、入力チャンネルの名称等を規定したネームライブラリ等が記憶されている。これらのデータを「ライブラリデータ」と呼ぶ。

【 0 0 1 4 】

1. 2. エンジン

次に、図 1 (b)を参照し、ミキシングシステムに使用されるエンジン 2 0 0 のハードウェア構成について説明する。図において 2 0 2 は信号処理部であり、DSP アレイによって構成されている。信号処理部 2 0 2 は「9 6」のモノラル入力チャンネルに対してミキシング処理を施し、その結果を「4 8」のモノラル出力チャンネル等に出力することが可能である。なお、信号処理部 2 0 2 において実行されるミキシング処理のアルゴリズムの詳細については後述する。

【 0 0 1 5 】

2 0 4 は波形 I / O 部であり、マイクあるいはラインレベルのアナログ信号をデジタル信号に変換する複数の A D コンバータと、信号処理部 2 0 2 から出力されたデジタル信号をアナログ信号に変換しアンプ等に供給する複数の D A コンバータと、外部機器から供給されたデジタル音声信号をエンジン 2 0 0 内で使用される所定フォーマットのデジタル信号に変換するとともにエンジン 2 0 0 内のデジタル音声信号のフォーマットを変換し外部機器に出力するデジタル入出力部とから構成されている。

【 0 0 1 6 】

2 0 6 はカスケード I / O 部であり、ここを介してエンジン 2 0 0 を他のエンジンにカスケード接続することにより、ミキシングシステムの処理能力を向上させることが可能である（詳細は後述する）。2 1 0 はデータ I / O 部であり、上記コンソール 1 0 0 のデータ I / O 部 1 1 0 との間でデジタル音声信号のやりとりを行う。また、2 1 2 は通信 I / O 部であり、コンソール 1 0 0 の通信 I / O 部 1 1 2 との間で制御信号のやりとりを行う。2 1 4 は表示器であり、エンジン 2 0 0 側の作業員に対して各種の情報を表示する。

【 0 0 1 7 】

2 1 6 はその他 I / O 部であり、各種の外部機器との間で音声信号等のやりとりを行う。2 1 8 は CPU であり、フラッシュメモリ 2 2 0 に格納された制御プログラムに基づいて、バス 2 2 4 を介してエンジン 2 0 0 内の各部を制御する。2 2 2 は RAM であり、CPU 2 1 8 のワークメモリとして使用される。

【 0 0 1 8 】

1. 3. ミキシングシステムの構成

1. 3. 1. シングルコンソールシステム

次に、上記コンソール 1 0 0 およびエンジン 2 0 0 によって構成可能なミキシングシステムの構成を図 2 (a)～(d)を参照し説明する。まず、同図(a)は 1 台のコンソール 1 0 0 と、 1 台のエンジン 2 0 0 とによって構成されたシングルコンソールシステムの構成例である。なお、図 2 においては、複数台のコンソール 1 0 0 およびエンジン 2 0 0 を区別するため、これらの符号に (A, B, C, ……) 等のアルファベットの符号を付す。

【 0 0 1 9 】

上述したようにコンソール 1 0 0 A は、「 4 8 」個の電動フェーダを有しており、エンジン 2 0 0 E においてはモノラル「 9 6 」入力チャンネルの処理が可能である。この「 9 6 」入力チャンネルは第 1 および第 2 レイヤに分割されており、例えば第 1 レイヤには第 1 ないし第 4 8 入力チャンネル、第 2 レイヤには第 4 9 ないし第 9 6 入力チャンネルが割り当てられる。また、操作子群 1 1 4 には、電動フェーダ部 1 0 4 によって操作すべきレイヤを選択するためのレイヤ選択スイッチが設けられている。

【 0 0 2 0 】

従って、オペレータは、入力チャンネルのレベル等を調節する場合には、当該入力チャンネルの属するレイヤをレイヤ選択スイッチによって選択し、しかる後に対応するフェーダを操作すればよい。フェーダが操作されると、カレント領域 1 2 2 a 内の対応する箇所に記憶されている操作量（減衰量）が更新される。そして、更新された箇所のデータがコンソール 1 0 0 A からエンジン 2 0 0 E に送信されることにより、信号処理部 2 0 2 内のアルゴリズムにおけるパラメータが変更され、出力される音声信号に対してフェーダ操作が反映されることになる。

【 0 0 2 1 】

また、オペレータによってシーンリコールの操作が行われると、シーン領域 1 2 2 b から指定されたシーンデータが読み出され、その内容がカレント領域 1 2 2 a に転送される。これによってカレント操作データの内容は大幅に変更されることになる。そして、フェーダ等が操作された場合と同様に、シーンリコールに

よって更新されたカレント操作データの内容はコンソール 1 0 0 A からエンジン 2 0 0 E に送信される。これにより、信号処理部 2 0 2 内のアルゴリズムにおいて、リコールされたシーンの内容が反映される。

【 0 0 2 2 】

1. 3. 2. デュアルコンソールシステム

上記シングルコンソールシステムにおいては、制御すべき入力チャンネルに応じてレイヤを選択する操作が必要であるが、かかる操作は煩雑であり、異なるレイヤに属する入力チャンネルを同時に制御することも困難になる。そこで、本実施形態においては、図 2 (b) に示すように、2 台のコンソールを用いることによってモノラル「9 6」入力チャンネルを同時に制御することが可能である。かかる構成をデュアルコンソールシステムという。

【 0 0 2 3 】

図 2 (b) において「2」台のコンソール 1 0 0 A, 1 0 0 B は、これらのデュアル I / O 部 1 0 6 を介して相互に接続されている。そして、コンソール 1 0 0 A のデータ I / O 部 1 1 0 および通信 I / O 部 1 1 2 は、エンジン 2 0 0 E のデータ I / O 部 2 1 0 および通信 I / O 部 2 1 2 に各々接続されている。このように、エンジン 2 0 0 E に直接的に接続されている側のコンソールを「マスタコンソール」といい、他方のコンソールを「スレーブコンソール」と呼ぶ。

【 0 0 2 4 】

これら「2」台のコンソールのうち一方の電動フェーダ部 1 0 4 に対して第 1 レイヤ、他方の電動フェーダ部 1 0 4 に対して第 2 レイヤを割り当てることにより、「9 6」入力チャンネルの各々に対して独立した電動フェーダを割り当てることが可能になる。ここで、デュアルコンソールシステムを構成する各コンソールのカレント領域 1 2 2 a には、シングルコンソールシステムの場合と同様のカレント操作データが各々記憶される。すなわち、各コンソールの電動フェーダ部 1 0 4 に割り当てられているレイヤにかかわらず、両コンソールのカレント領域 1 2 2 a には、「9 6」入力チャンネルの各々に対して減衰量等のパラメータが記憶されることになる。

【 0 0 2 5 】

そして、デュアルコンソールシステムにおいては、コンソール 1 0 0 A, 1 0 0 B における各カレント領域 1 2 2 a の内容は、同一になるように制御される。例えば、一方のコンソールにおいてなんらかの操作が行われると、当該コンソールにおけるカレント操作データは当該操作に応じて更新される。この更新内容は他方のコンソールに送信され、他方のコンソールにおいてもカレント操作データが同様に更新されるのである。

【 0 0 2 6 】

そして、最終的にエンジン 2 0 0 E に対して各種パラメータを送信するコンソールは必ずマスタコンソール 1 0 0 A である。換言すれば、エンジン 2 0 0 E 内のアルゴリズムにおけるパラメータは、コンソール 1 0 0 A のカレント操作データに応じて設定され、コンソール 1 0 0 B 内のカレント操作データは関知されないことになる。

【 0 0 2 7 】

ここで、一方のコンソールにおいてシーンリコール操作があった場合の対処方法が問題になる。シーンリコール操作が行われたコンソールから他方のコンソールに対してシーンの内容を全て送信すると、伝送すべきデータ量が多大になるため、双方のコンソールにシーンリコールを反映するまでに要する時間が長くなりすぎる。これを防止するため、本実施形態においては、シーンリコール操作のみ（すなわちどのシーンがリコールされたか）が相互に伝達され、実際のシーンの再現は、各々のコンソールにおけるシーンデータの内容に基づいて実行される。このため、各コンソールにおけるシーン領域 1 2 2 b の内容は、基本的には、予め一致させておく必要がある。

【 0 0 2 8 】

1. 3. 3. シングルコンソールシステムのカスケード接続

上記シングルコンソールシステムにおいて、入力チャンネルの総数「96」そのものが不足している場合には、図 2 (c) に示すように、2 台のコンソールおよび 2 台のエンジンを用いることによって 2 倍の入力チャンネルを確保することが可能である。図 2 (c) において、コンソール 1 0 0 A およびエンジン 2 0 0 E はこれらの各 I / O 部 1 1 0, 1 1 2, 2 1 0, 2 1 2 を介して相互に接続されて

いる。また、コンソール 1 0 0 B およびエンジン 2 0 0 F も同様に接続されている。

【 0 0 2 9 】

そして、エンジン 2 0 0 E, 2 0 0 F は、これらのカスケード I / O 部 2 0 6 を介して相互に接続されている。このようなエンジン 2 0 0 E, 2 0 0 F の接続方法をカスケード接続という。かかる構成において、コンソール 1 0 0 A, 1 0 0 B のカレント操作データは独立しており、各々のコンソールにおいて各「 9 6 」入力チャンネルが制御されることになる。また、シーンの切換等を両コンソール間でリンクさせるか否かはオペレータによって指定可能である。

【 0 0 3 0 】

1. 3. 4. デュアルコンソールシステムのカスケード接続

また、「 2 」組のデュアルコンソールシステムに対してカスケード接続を行うことも可能である。かかる場合の構成例を図 2 (d) に示す。図においてコンソール 1 0 0 A, 1 0 0 B およびエンジン 2 0 0 E は図 2 (b) と同様にデュアルコンソールシステムを構成している。また、コンソール 1 0 0 C, 1 0 0 D およびエンジン 2 0 0 F も同様にデュアルコンソールシステムを構成している。そして、エンジン 2 0 0 E, 2 0 0 F は、これらのカスケード I / O 部 2 0 6 を介して相互に接続されている。

【 0 0 3 1 】

2. 実施形態のアルゴリズム構成

2. 1. ミキシングシステムのアルゴリズム

2. 1. 1. シングルコンソールシステム

次に、シングルコンソールシステム（図 2 (a)）において信号処理部 2 0 2 等によって実現されるミキシング処理のアルゴリズムの構成を図 4 を参照し説明する。図において 2 3 2 はアナログ入力部であり、複数チャンネルのアナログ音声信号をデジタル信号に変換する。また、2 3 4 はデジタル入力部であり、外部から供給された複数チャンネルのデジタル音声信号をエンジン 2 0 0 内で使用される所定フォーマットのデジタル信号に変換する。これら入力部 2 3 2, 2 3 4 は波形 I / O 部 2 0 4 によって実現される。

【 0 0 3 2 】

次に、2 3 6 は内蔵エフェクタであり、最大「8」チャンネルの音声信号に対してエフェクト処理を施す。また、2 3 8 は内蔵イコライザであり、周波数特性等のイコライジング処理を最大「2 4」チャンネルの音声信号に対して施すことができる。2 4 2 は入力チャンネル調整部であり、コンソール 1 0 0 A における操作に基づいて、最大「9 6」チャンネルの入力チャンネルに対して音量・音質等の調整を行う。

【 0 0 3 3 】

2 4 0 は入力パッチ部であり、上記各入力部 2 3 2、2 3 4、内蔵エフェクタ 2 3 6 または内蔵イコライザ 2 3 8 から供給されたデジタル音声信号を入力チャンネル調整部 2 4 2 の任意のチャンネルに割り当てる。但し、アナログ入力部 2 3 2 から入力された所定の「1」チャンネルは、エンジン 2 0 0 E 側の作業員の音声信号を伝送するコムイン信号 COMM_IN_1 として、後述するモニタ系統を介して、コンソール 1 0 0 A 側に送信される。

【 0 0 3 4 】

2 4 4 はミキシングバスであり、入力チャンネル調整部 2 4 2 を介して音量・音質調節されたデジタル音声信号を最大「4 8」系統のモノラル音声信号にミキシングする。2 5 4 は出力チャンネル調整部であり、これら「4 8」系統のモノラル音声信号に対して、音量調節等を行う。なお、「4 8」系統のミキシングバス 2 4 4 と出力チャンネルを、それぞれ所定の「2」系統づつペア設定することが可能であり、ペア設定された系統ではステレオ音声信号のミキシングが行われる。

【 0 0 3 5 】

次に、2 5 6 はマトリクス出力チャンネル部であり、出力チャンネル調整部 2 5 4 における「4 8」系統のミキシング結果をさらにミキシングし出力する。マトリクス出力チャンネル部 2 5 6 においては、モノラル「2 4」系統の音声信号をミキシングすることが可能になっている。そして、各出力チャンネル部 2 5 4、2 5 6 におけるミキシング結果は、出力パッチ部 2 5 8 に供給される。

【 0 0 3 6 】

次に、260はアナログ出力部であり、供給されたデジタル音声信号をアナログ信号に変換する。これらアナログ信号は、コンサートホール内への放音、レコーディング等のためにアンプあるいはレコーディング機材等（図示せず）に供給される。また、262はデジタル出力部であり、供給されたデジタル音声信号のフォーマットを変換し、デジタル・レコーディング機材等（図示せず）に供給する。これら出力部260、262は波形I/O部204によって実現される。

【0037】

出力パッチ部258は、各出力チャンネル部254、256から出力されたデジタル音声信号を各出力部260、262における任意のチャンネルに割り当てる。ここで、必要な場合には、これらデジタル音声信号の一部を内蔵エフェクタ236あるいは内蔵イコライザ238への入力に割り当てることも可能である。従って、ある出力チャンネルに対してエフェクト処理／イコライジング処理を施した結果を再び入力パッチ部240に戻し、これを新たな入力チャンネルの信号として用いることができる。

【0038】

また、出力パッチ部258には、トークバックアウトスイッチ257を介して、一または複数のオペレータの音声等であるトークバック信号TB_OUTが入力される。トークバック信号TB_OUTは、機器のセッティング時にはアナログ出力部260を介してコンサートホールに放音される。これにより、オペレータ自身の音声によってコンサートホールの音響テストを行い、あるいは舞台上の作業員に対する放送を行うことができる。また、コンサートの本番時には、トークバックアウトスイッチ257はオフ状態に設定され、トークバック信号TB_OUTはエンジン200E側の作業員に対する通話に用いられる。

【0039】

次に、250はモニタ用セクタであり、オペレータの操作に基づいて上述した系統中の任意の箇所を選択する。すなわち、コンソール100にはモニタ用セクタ250の選択状態を設定するためのモニタスイッチが設けられている。また、252は他のモニタ用セクタであり、シングルコンソールシステムにおいては、オペレータはモニタ用セクタ250、252の双方の選択状態を任意に

設定することが可能である。これらセレクタ 2 5 0, 2 5 2 によって選択された信号は、第 1, 第 2 モニタ信号 MON 1, MON 2 として出力される。

【 0 0 4 0 】

また、各コンソールにおける各フェーダの近傍には、該フェーダに対応するデジタル音声信号を監視するか否かを指定するキュースイッチが設けられている。2 4 6 はキューバスであり、キュースイッチがオンされた箇所のデジタル音声信号をミキシングし、第 1 キュー信号 CUE 1 として出力する。

【 0 0 4 1 】

なお、第 1, 第 2 モニタ信号 MON 1, MON 2 は、主としてコンサートホール等に放音されている音声信号のモニタ等のために用いられ、第 1 キュー信号 CUE 1 は主として一または複数の特定の入力チャンネルあるいは出力チャンネル等をモニタするために用いられることが多い。これらの信号は、後述するモニタ系統を介して、コンソール 1 0 0 側に送信される。

【 0 0 4 2 】

なお、本明細書において、コンソール 1 0 0 内における信号の称呼はエンジン 2 0 0 内における称呼とは異なる。すなわち、コンソール 1 0 0 においてモニタ可能な信号は、「モニタ信号 MON_A, MON_B」および「キュー信号 CUE」である。シングルコンソールシステムにおいては、モニタ信号 MON_A, MON_B は各々第 1, 第 2 モニタ信号 MON 1, MON 2 に等しく、キュー信号 CUE は第 1 キュー信号 CUE 1 に等しい。

【 0 0 4 3 】

2. 1. 2. デュアルコンソールシステム

次に、デュアルコンソールシステム（図 2 (b)）において信号処理部 2 0 2 等によって実現されるアルゴリズムの構成を説明する。かかる場合のアルゴリズムは上述したシングルコンソールシステム（図 4）と同様であるが、以下に述べる点が異なる。

まず、デュアルコンソールシステムにおいては、キューバス 2 4 6 に加えて、破線で示す追加のキューバス 2 4 8 が設けられる。そして、キューバス 2 4 6 においてはマスタコンソール 1 0 0 A のキュースイッチに基づいて第 1 キュー信号

CUE 1 が合成され、キューバス 2 4 8 においてはスレーブコンソール 1 0 0 B のキュースイッチに基づいて第 2 キュー信号 CUE 2 が合成される。

【 0 0 4 4 】

そして、第 1 キュー信号 CUE 1 はコンソール 1 0 0 A におけるキュー信号 CUE として用いられ、第 2 キュー信号 CUE 2 はコンソール 1 0 0 B におけるキュー信号 CUE として用いられる。これにより、コンソール 1 0 0 A, 1 0 0 B のオペレータは、各々自己がコントロールするコンソールのキュースイッチの操作に応じて、独立したキュー信号 CUE をモニタすることができる（後述するキュー・リンク・スイッチ 1 4 9 がオフの場合）。一方、一人のオペレータがコンソール 1 0 0 A, 1 0 0 B の両方を操作する場合には、キュー・リンク・スイッチ 1 4 9 をオンにすることにより、何れか一方のコンソールで行われたキュースイッチの操作が他方のコンソールに伝送される。これにより、第 1 キュー信号 CUE 1 および第 2 キュー信号 CUE 2 として、同じキュースイッチ操作に応じた信号が選択され、両コンソールにおいて同一のキュー信号 CUE をモニタできるようになる。

【 0 0 4 5 】

さらに、エンジン 2 0 0 E 側の作業員がコンソール 1 0 0 A, 1 0 0 B の双方のオペレータに対して独立して音声信号を送信するために、アナログ入力部 2 3 2 から入力された所定の「2」チャンネルがコムイン信号 COMM_IN_1, COMM_IN_2 に割り当てられる。一方、両コンソール 1 0 0 A, 1 0 0 B からのトークバック信号はミキシングされた後にトークバック信号 TB_OUT として出力パッチ部 2 5 8 に供給される。出力パッチ部 2 5 8 においては、トークバック信号 TB_OUT が上記作業員へ送信されるようにパッチングされる。このため、本実施形態においては、デュアルコンソールシステムにおいても、トークバック信号 TB_OUT は「1」系統である。「1」系統の方が無駄が少ないのであるが、トークバック信号 TB_OUT を「2」系統にして、各コンソールから作業員へ別々に信号を送信するようにしてもよい。

【 0 0 4 6 】

また、モニタ用セレクト 2 5 0 はコンソール 1 0 0 A 内のモニタスイッチのみ

によって選択状態が設定され、モニタ用セクタ 2 5 2 はコンソール 1 0 0 B 内のモニタスイッチのみによって選択状態が設定される。また、モニタ用セクタ 2 5 0 によって選択された第 1 モニタ信号 MON 1 はマスタコンソール 1 0 0 A にモニタ信号 MON_A として供給されるとともに、スレーブコンソール 1 0 0 B にモニタ信号 MON_B として供給される。

【 0 0 4 7 】

逆に、モニタ用セクタ 2 5 2 によって選択された第 2 モニタ信号 MON 2 はマスタコンソール 1 0 0 A にモニタ信号 MON_B として供給されるとともに、スレーブコンソール 1 0 0 B にモニタ信号 MON_A として供給される。かかるアルゴリズムは、コンソール 1 0 0 A, 1 0 0 B のオペレータ側から見ると次のようになる。すなわち、オペレータは、自己が管理するコンソールにおいてモニタスイッチを操作すると、その結果が必ずモニタ信号 MON_A に反映される。また、キュースイッチを操作すると、その結果が必ずキュー信号 CUE に反映される。さらに、他方のコンソールにおけるモニタスイッチの操作はモニタ信号 MON_B に反映される。

【 0 0 4 8 】

このように、本実施形態においては、デュアルコンソールシステムにおいて、コンソール 1 0 0 A, 1 0 0 B におけるキュー、モニタ系統の独立性を保持しつつ、これらコンソールにおける操作に統一性、互換性を確保することができる。これにより、オペレータによるキュー、モニタ系統の操作ミスを著しく削減することができ、仮に一方のオペレータにおいて誤操作が発生した場合においても、他方のオペレータに及ぶ影響を最小限に止めることが可能である。

【 0 0 4 9 】

但し、デュアルコンソールシステムにおいても、オペレータの設定によってキューバスを一系統のみ（2 4 6 のみ）とすることが可能である。これは、双方のコンソールを一人のオペレータが操作する場合には、キュー信号を一系統のみにしておく方が操作上便利だからである。すなわち、後述するキュー・リンク・スイッチ 1 4 9（図 3 参照）によって、オペレータはキュー信号の系統数を「1」または「2」に切り換えることができる。キュー信号の系統数が「1」に設定さ

れると、マスタまたはスレーブコンソールの何れか一方において押下されたキュースイッチに基づく音声信号が全てキューバス 2 4 6 においてミキシングされ、その結果が同一内容の第 1, 第 2 キュー信号 CUE 1, CUE 2 として両コンソールに供給される。

【 0 0 5 0 】

2. 1. 3. システムのカスケード接続

2 系統のシングルコンソールシステムまたはデュアルコンソールシステムのエンジン 2 0 0 E, 2 0 0 F 同士をカスケード接続した場合のアルゴリズムは、原則的には図 4 の構成を 2 系統設け、双方のミキシングバス 2 4 4 およびキューバス 2 4 6, 2 4 8 をリンクした構成に等しくなる。ここで、これらバスリンクの詳細を図 5 を参照し説明する。なお、図 5 においてエンジン 2 0 0 E 内で実行されるアルゴリズムの各部の符号は図 4 に示す符号に「e」を付し、エンジン 2 0 0 F 内で実行されるアルゴリズムの各部の符号は図 4 に示す符号に「f」を付したものである。

【 0 0 5 1 】

図 5 において、エンジン 2 0 0 E 側のミキシングバス 2 4 4 e と出力チャンネル調整部 2 5 4 e の間には、遅延回路 2 6 4 e および加算器 2 6 6 e が順次介挿されている。同様に、エンジン 2 0 0 F 側のミキシングバス 2 4 4 f と出力チャンネル調整部 2 5 4 f の間には、遅延回路 2 6 4 f および加算器 2 6 6 f が順次介挿されている。そして、ミキシングバス 2 4 4 e におけるミキシング結果は加算器 2 6 6 f に供給され、ミキシングバス 2 4 4 f におけるミキシング結果は加算器 2 6 6 e に供給される。

【 0 0 5 2 】

なお、遅延回路 2 6 4 e, 2 6 4 f および加算器 2 6 6 e, 2 6 6 f は各「1」系統のみ図示しているが、これらは各々「 4.8×2 」のミキシングチャンネル毎に設けられている。これにより、出力チャンネル調整部 2 5 4 e, 2 5 4 f に供給される信号は、共にミキシングバス 2 4 4 e, 2 4 4 f のミキシング結果をさらにミキシングした結果になり、出力チャンネル調整部 2 5 4 e, 2 5 4 f に供給される信号は両エンジン 2 0 0 E, 2 0 0 F において等しい信号になる。こ

れにより、カスケード接続時においては、2つのコンソールシステムを合せての入力チャンネル数は「192」であり、それを「48」本のバスを介してミキシングし、各コンソール毎に対応する「48」の出力チャンネルで調整出力するようなミキシングシステムが構築される。

【0053】

また、エンジン200E側のキューバス246eの出力は遅延回路270eおよび加算器272eを順次介して第1キュー信号CUE1(E)として出力され、エンジン200F側のキューバス246fの出力は遅延回路270fおよび加算器272fを順次介して第1キュー信号CUE1(F)として出力される。そして、キューバス246eのミキシング結果はスイッチ274fを介して加算器272fに供給され、キューバス246fのミキシング結果はスイッチ274eを介して加算器272eに供給される。

【0054】

ここで、スイッチ274e, 274fをオン状態に設定すると、エンジン200E, 200Fにおける第1キュー信号CUE1(E), (F)は等しくなり、スイッチ274e, 274fをオフ状態に設定すると、両第1キュー信号CUE1(E), (F)は独立した信号になる。これは、カスケード接続された2台のエンジンに対応するコンソールを一人のオペレータが操作する場合には、キュー信号を一系統のみにしておく方が操作上便利であり、別々のオペレータが各コンソールを操作する場合にはキュー信号を独立して選択できるように設定することが望ましいからである。なお、キューバスのリンク構成が図5のように設定されているため、スイッチ274e, 274fをオン状態にした場合、2つのシステムの何れかにおいてオンされたキュースイッチによるキュー信号を、両方のシステムでモニタすることができる。但し、この場合においても、キュースイッチの操作はカスケード接続された2つのシステム間ではリンクされない。

【0055】

また、デュアルコンソールシステムがカスケード接続され、双方のエンジンにおいて第2キュー信号CUE2用のキューバス248e, 248fが形成される場合には、これらキューバス248e, 248fに対しても同様のアルゴリズム

が設定される。すなわち、エンジン 2 0 0 E 側のキューバス 2 4 8 e の出力は遅延回路 2 7 6 e および加算器 2 7 8 e を順次介して第 2 キュー信号 C U E 2 (E) として出力され、エンジン 2 0 0 F 側のキューバス 2 4 8 f の出力は遅延回路 2 7 6 f および加算器 2 7 8 f を順次介して第 2 キュー信号 C U E 2 (F) として出力される。そして、キューバス 2 4 8 e のミキシング結果はスイッチ 2 8 0 f を介して加算器 2 7 8 f に供給され、キューバス 2 4 8 f のミキシング結果はスイッチ 2 8 0 e を介して加算器 2 7 8 e に供給される。

【 0 0 5 6 】

ところで、図 5 においては、何れのエンジンにおいても、自機側で生成された信号は遅延回路を介して遅延されるのに対して、カスケード接続の相手側から受信した信号は遅延されない点に特徴がある。例えば、ミキシングバス 2 4 4 e におけるミキシング結果は遅延回路 2 6 4 e を介して自機側の出力チャンネル調整部 2 5 4 e に供給されるのに対して、該ミキシング結果はなんら遅延回路を通過することなく、加算器 2 6 6 f を介して相手側の出力チャンネル調整部 2 5 4 f に供給される。

【 0 0 5 7 】

これは、エンジン 2 0 0 E, 2 0 0 F 間における伝送遅延を補償するためである。例えば、ミキシングバス 2 4 4 e におけるミキシング結果は、実際にはエンジン 2 0 0 E 側の信号処理部 2 0 2 e からカスケード I / O 部 2 0 6 e、ケーブル、エンジン 2 0 0 F 側のカスケード I / O 部 2 0 6 f を順次介して信号処理部 2 0 2 f に供給されるため、伝送遅延が生ずることを避けることができない。この遅延した信号と、ミキシングバス 2 4 4 f におけるミキシング結果とを単にミキシングすると、位相ずれなどの不具合が発生する。そこで、遅延回路 2 6 4 f を介して、かかる伝送遅延と同等の遅延時間をミキシングバス 2 4 4 f のミキシング結果に付与しておくことにより、位相ずれなどの無いミキシング結果を得ることができるのである。すなわち、各コンソールシステムの「 4 8 」チャンネルの出力チャンネル調整部 2 5 4 e, 2 5 4 f には、各「 4 8 」本のミキシングバス 2 4 4 e, 2 4 4 f のミキシング結果の位相を相互に揃えてミキシングした「 4 8 」のミキシング結果が供給され、各ミキシング結果に対して両方のコンソー

ルシステムで異なる調整を施して出力することができるようになっている。

【0058】

2. 2. モニタ系統のアルゴリズム

2. 2. 1. アルゴリズムの内容

次に、本実施形態におけるモニタ系統のアルゴリズムについて図6、図7を参照し説明する。なお、ここではデュアルコンソールシステムのカスケード接続（図2(d)）の場合についてのみ説明する。これは、同システムのモニタ系統が最大規模の系統になり、他のシステムにおいては不要な部分を無視すれば足りるからである。

【0059】

図6において、300e、302eはトークバック入力スイッチであり、コンソール100A、100Bに設けられたオン／オフスイッチ（図示せず）の操作状態に基づいて、エンジン200Eに供給されたトークバック信号TB_A、TB_Bのオン／オフ状態を切り換える。また、コンソール100A、100Bの内部において152a、152bはモニタアンプであり、入力スイッチ300e、302eのオン／オフ状態に基づいてゲインが増減される。

【0060】

ここで、モニタアンプ152a、152bにおけるゲイン調節の必要性について説明しておく。モニタアンプ152a、152bを介して出力された各コンソールのモニタ信号MON_Aがモニタスピーカを介して放音される場合、そのモニタ音がトークバックマイクを介して回り込み、雑音が発生する場合がある。これを防止するために、モニタアンプ152a、152bにおいては、トークバックの際にモニタ音の音量が減衰されるのである。かかる動作を「トークバック・ディマ」と呼ぶ。

【0061】

なお、オペレータがモニタ音をヘッドフォンを介してモニタする場合にはトークバック・ディマは不要であるため、トークバック・ディマを有効にするか否か、また有効にする場合の減衰量はコンソール100A、100B側にて自在に設定することが可能である。また、マスタコンソール100Aにおいては、スイッ

チ 1 5 4 a によって、コンソール 1 0 0 A, 1 0 0 B のトークバック・ディマを連動させるか否かが設定される。例えば、コンソール 1 0 0 A, 1 0 0 B が物理的に近接して配置され、各々のオペレータがモニタスピーカを用いてモニタを行っている場合には、一方のコンソールの側のモニタ音が他方のトークバックマイクを介して回り込むことがある。そこで、かかる場合には、少なくとも一方のコンソールにおいてトークバック・ディマが実行される場合には、他方のコンソールにおいても必ず実行されるように連動させることが好適である。

【 0 0 6 2 】

セクタ 2 5 0 (図 4 参照) から出力された第 1 モニタ信号 MON 1 は、アンプ 3 0 6 e, 加算器 3 1 0 e, 3 1 2 e を順次介して、コンソール 1 0 0 A のモニタ信号 MON_A として出力される。また、入力スイッチ 3 0 2 e を介して出力されるトークバック信号 TB_B は、スイッチ 3 0 4 e を介して該加算器 3 1 0 e に供給される。従って、スイッチ 3 0 4 e をオン状態にすると、コンソール 1 0 0 B からのトークバック信号 TB_B が第 1 モニタ信号 MON 1 にミキシングされコンソール 1 0 0 A に供給される。

【 0 0 6 3 】

同様に、セクタ 2 5 2 から出力された第 2 モニタ信号 MON 2 は、アンプ 3 2 6 e, 加算器 3 3 0 e, 3 3 2 e を順次介して、コンソール 1 0 0 B のモニタ信号 MON_A として出力される。また、入力スイッチ 3 0 0 e を介して出力されるトークバック信号 TB_A は、スイッチ 3 2 4 e を介して該加算器 3 3 0 e に供給される。従って、スイッチ 3 2 4 e をオン状態にすると、コンソール 1 0 0 A からのトークバック信号 TB_A が第 2 モニタ信号 MON 2 にミキシングされコンソール 1 0 0 B に供給される。

【 0 0 6 4 】

これらスイッチ 3 0 4 e, 3 2 4 e は、コンソール 1 0 0 A, 1 0 0 B が相互に物理的に離れている場合にオン状態にすると好適である。これにより、両コンソールのオペレータは、双方のトークバック信号およびモニタ信号 MON_A を用いて会話することが可能になるからである。

【 0 0 6 5 】

また、エンジン 2 0 0 E におけるコムイン信号 COMM_IN_1 (E) は、加算器 3 1 4 e、スイッチ 3 1 6 e を介してゲート回路 3 1 8 e に供給される。従って、コムイン信号を聞く必要が無い場合には、オペレータがスイッチ 3 1 6 e をオフ状態に設定しておくといよい。また、ゲート回路 3 1 8 e においては、供給されたコムイン信号のレベルが所定の閾値以上になると、当該コムイン信号を加算器 3 1 2 e に供給するとともに、該コムイン信号のレベルが該閾値未満であれば該コムイン信号を遮断する。

【 0 0 6 6 】

これにより、例えばコムイン信号用のマイクを介して低レベルのノイズがゲート回路 3 1 8 e に供給されたとしても、これがオペレータに聞こえることが無いため、オペレータのモニタリング作業に支障を来すことが無い。一方、エンジン 2 0 0 E 側の作業員がある程度大きな音声でコムイン信号を入力すると、ゲート回路 3 1 8 e が導通状態になり、コムイン信号 COMM_IN_1 (E) が第 1 モニタ信号 MON 1 にミキシングされるから、作業員の音声を的確にコンソール 1 0 0 A のオペレータに伝達することができる。

【 0 0 6 7 】

また、加算器 3 1 4 e には、カスケード接続の相手側であるエンジン 2 0 0 F に接続されたマスタコンソール 1 0 0 C のトークバック信号 TB_C がスイッチ 3 2 2 e を介して供給され、スレーブコンソール 1 0 0 D のトークバック信号 TB_D がスイッチ 3 2 0 e を介して供給され、さらにエンジン 2 0 0 F におけるコムイン信号 COMM_IN_1 (F) がスイッチ 3 0 8 e を介して供給される。従って、スイッチ 3 0 8 e、3 2 0 e、3 2 2 e のうち任意の一または複数のスイッチをオン状態に設定すると、これに対応してコムイン信号 COMM_IN_1 (F)、トークバック信号 TB_D またはトークバック信号 TB_C が第 1 モニタ信号 MON 1 にミキシングされ、コンソール 1 0 0 A のオペレータに聴取されることになる。

【 0 0 6 8 】

ここで、アンプ 3 0 6 e のゲインは、ゲート回路 3 1 8 e に連動している。すなわち、ゲート回路 3 1 8 e が導通状態になると、アンプ 3 0 6 e のゲインが自

動的に低下する。これにより、モニタ信号音等に妨げられることなく、コムイン信号を的確にオペレータに伝達することができる。

【 0 0 6 9 】

上述した構成と同様に、コムイン信号COMM_IN_2 (E) は、加算器334 e、スイッチ336 e、ゲート回路338 eを介して加算器332 eに供給されるから、コムイン信号COMM_IN_2 (E) を第2モニタ信号MON2にミキシングすることが可能である。さらに、コンソール100C, 100Dのトークバック信号TB_C, TB_Dおよびエンジン200Fのコムイン信号COMM_IN_2 (E) が各々スイッチ342 e, 340 eおよびスイッチ328 eを介して加算器334 eに供給されるから、これらのスイッチをオン状態に設定すると、対応するトークバック信号が第2モニタ信号MON2にミキシングされ、コンソール100Bのオペレータに聴取されることになる。

【 0 0 7 0 】

また、トークバック信号TB_Aは加算器352 eを介して、スイッチ356 eの第1の入力端に供給される。トークバック信号TB_Bは、加算器362 eを介して、スイッチ356 eの第2の入力端に供給される。そして、トークバック信号TB_A, TB_Bは、加算器352 e, 362 eおよび364 eを介してミキシングされ、スイッチ356 eの第3の入力端に供給される。そして、スイッチ356 eにおいては、これら第1～第3の入力端に供給された信号のうち一の信号が選択される。

【 0 0 7 1 】

また、354 eは発振器であり、コンサートホール等の音響状態をテストするための正弦波信号等を出力する。発振器354 eの出力信号またはスイッチ356 eにおいて選択されたトークバック信号のうち一方の信号がスイッチ358 eにおいて選択され、選択された信号がエンジン200E用のトークバック信号TB_OUT (E) として出力され、上述したようにエンジン200Eの出力パッチ部258 (図4参照) に供給される。なお、上述したように、「2」系統のトークバック信号TB_OUTの双方を出力パッチ部258に供給するようにしてもよい。

【 0 0 7 2 】

ここで、スイッチ 3 5 8 e の切換状態は、スイッチ 3 5 6 e および入力スイッチ 3 0 0 e, 3 0 2 e の状態に応じて自動的に設定される。すなわち、スイッチ 3 5 6 e が第 1 の入力端に切り換えられている場合には入力スイッチ 3 0 0 e がオン状態になった時に、スイッチ 3 5 6 e が第 2 の入力端に切り換えられている場合には入力スイッチ 3 0 2 e がオン状態になった時に、また、スイッチ 3 5 6 e が第 3 の入力端に切り換えられている場合には入力スイッチ 3 0 0 e, 3 0 2 e の何れかがオン状態になった時に、スイッチ 3 5 8 e はスイッチ 3 5 6 e 側に切り換えられ、上記以外の場合はスイッチ 3 5 8 e は発振器 3 5 4 e 側に切り換えられる。

【 0 0 7 3 】

これにより、トークバック信号 TB_A, TB_B のうち何れかがスイッチ 3 5 6 e を介して出力される場合には、スイッチ 3 5 8 e は必ずスイッチ 3 5 6 e 側に切り換えられ、トークバック信号 TB_OUT にはトークバック信号 TB_A, TB_B の少なくとも一方がミキシングされることになる。また、加算器 3 5 2 e には、スイッチ 3 6 0 e を介してトークバック信号 TB_C が供給され、加算器 3 6 2 e にはスイッチ 3 6 6 e を介してトークバック信号 TB_D が供給される。従って、スイッチ 3 6 0 e, 3 6 6 e のうち一方または双方をオン状態にすることにより、トークバック信号 TB_C, TB_D をミキシングしたトークバック信号 TB_OUT (E) を出力することができる。

【 0 0 7 4 】

次に、3 5 0 e, 3 6 8 e はトークバック・ディマ連動制御用のスイッチである。スイッチ 3 5 0 e がオン状態に設定されると、エンジン 2 0 0 F 側のマスタコンソール 1 0 0 C においてトークバック・ディマが実行される場合には、エンジン 2 0 0 E 側のマスタコンソール 1 0 0 A においても連動してトークバック・ディマが実行される。また、スイッチ 3 6 8 e がオン状態に設定されると、エンジン 2 0 0 F 側のスレーブコンソール 1 0 0 D においてトークバック・ディマが実行される場合には、エンジン 2 0 0 E 側のスレーブコンソール 1 0 0 B においても連動してトークバック・ディマが実行される。

【 0 0 7 5 】

以上、図 6 を参照し主としてコンソール 1 0 0 A, 1 0 0 B およびエンジン 2 0 0 E において実行されるモニタシステムのアルゴリズムについて説明したが、コンソール 1 0 0 C, 1 0 0 D およびエンジン 2 0 0 F においても同様のアルゴリズムが実行される。その内容を図 7 に示す。図 7 において図 6 の各部に対応する部分には、各部の符号の末尾の文字「a」, 「b」または「e」を「c」, 「d」または「f」に変更した符号を付す。但し、コンソール 1 0 0 A, 1 0 0 D 間の通話経路に関連するスイッチの符号を 3 2 0 e, 3 2 0 f とし、コンソール 1 0 0 B, 1 0 0 C 間の通話経路に関連するスイッチの符号を 3 4 2 e, 3 4 2 f としている。

【 0 0 7 6 】

これらスイッチのうちスイッチ 1 5 4 a と 1 5 4 c のペア、スイッチ 3 0 4 e と 3 0 4 f のペア、スイッチ 3 2 4 e と 3 2 4 f のペアは、何れも連動しない。これらのスイッチは、対応するデュアルコンソールを構成する 2 台のコンソールの物理的な設置状態に応じて独自に設定することが好適だからである。

【 0 0 7 7 】

一方、スイッチ 3 0 8 e と 3 0 8 f のペア、スイッチ 3 2 0 e と 3 2 0 f のペア、スイッチ 3 2 2 e と 3 2 2 f のペア、スイッチ 3 2 8 e と 3 2 8 f のペア、スイッチ 3 4 0 e と 3 4 0 f のペア、スイッチ 3 4 2 e と 3 4 2 f のペア、スイッチ 3 5 0 e と 3 5 0 f のペア、スイッチ 3 6 0 e と 3 6 0 f のペア、スイッチ 3 6 6 e と 3 6 6 f のペア、およびスイッチ 3 6 8 e と 3 6 8 f のペアは何れも連動する。なお、これらのスイッチのオン／オフ状態は、対応するコンソールの何れにおいても操作可能である。

【 0 0 7 8 】

また、スイッチ 3 6 0 e, 3 6 0 f またはスイッチ 3 6 6 e, 3 6 6 f がオン状態になると、カスケード接続された相手側のトークバック信号がスイッチ 3 5 6 e を介して出力される場合には、スイッチ 3 5 8 e は自動的にスイッチ 3 5 6 e 側に切り換えられる。例えば、スイッチ 3 6 0 e, 3 6 0 f がオン状態にされ、かつ、スイッチ 3 5 6 e の接点が第 1 または第 3 の入力端に設定されていると

、トークバック信号TB_C用の入力スイッチ300fがオン状態にされた時にスイッチ358eはスイッチ356e側に自動的に切り換えられる。

【0079】

同様に、スイッチ366e, 366fがオン状態にされ、かつ、スイッチ356eの接点が第2または第3の入力端に設定されていると、トークバック信号TB_D用の入力スイッチ302fがオン状態にされた時にスイッチ358eはスイッチ356e側に自動的に切り換えられる。また、同様の動作がエンジン200F内においても実行される。

【0080】

2. 2. 2. ミキサの配置に応じたアルゴリズムのセッティング

次に、図8(a)～(e)を参照し、各コンソールの配置関係と、上記各スイッチの好適な設定状態について説明する。まず、同図(a)に示すようにカスケード接続の一方のグループ（カスケードグループ）を成すコンソール100A, 100Bを近接させ、他方のカスケードグループを成すコンソール100C, 100Dを近接させ、これらカスケードグループ間の距離を離すような配置状態が考えられる。また、同図(b)に示すように全てのコンソール100A～100Dを近接して配置することも考えられる。

【0081】

また、同図(c)のように、各カスケードグループのマスタであるコンソール100A, 100Cを近接配置し、スレーブコンソールであるコンソール100B, 100Dを近接配置し、マスタおよびスレーブコンソールの間の距離を離すような配置状態が考えられる。また、同図(d)のように全てのコンソール間の距離を離すような配置も可能であり、同図(e)に示すようにコンソール100A, 100Dおよびコンソール100B, 100Cを相互に近接させるような配置も可能である。

【0082】

同図(a)の例にあっては、スイッチ154a, 154cを共にオン状態に設定し、各カスケードグループ毎にトークバック・ディマを連動させるとよい。また、スイッチ304e, 304f, 324e, 324fをオフ状態に設定し、近接

するオペレータ同士はシステムを介さずに直接的に会話するようにするとよい。

【0083】

また、スイッチ350e, 350f, 368e, 368fをオフ状態に設定し、離れたコンソールによってトークバック・ディマが生じないようにするとよい。そして、スイッチ322e, 322f, 320e, 320f, 342e, 342f, 340e, 340fをオン状態に設定することにより、離れたコンソール間に通話経路を確保することが望ましい。さらに、スイッチ360e, 360f, 366e, 366fをオン状態に設定することにより、一方のエンジンにおけるトークバック信号TB_OUTに対して他方のエンジンにおけるトークバック信号をミキシングすることができ、これによってトークバック信号を統一化することができる。

【0084】

また、同図(b)に示すように全てのコンソール100A~100Dが近接配置される場合には、スイッチ154a, 154cをオン状態に設定し、スイッチ304e, 304f, 324e, 324fをオフ状態に設定するとよい。但し、スイッチ320e, 320f, 342e, 342fをオン状態に設定することにより、若干離れたコンソール100A, 100D間においてコミュニケーション経路を確保すると好適である。

【0085】

他の配置方法においても、同様の思想によって各スイッチのオン／オフ状態を決定すると好適である。すなわち、相互に近接したコンソールはトークバック・ディマを連動させるとともに、通話経路のスイッチをオフ状態にするとよい。また、相互に距離が離れたコンソール同士についてはトークバック・ディマを独立させるとともに、トークバック信号を用いた通話経路を形成するとよい。

【0086】

2. 3. コンソール上の操作子の構成

コンソール100内の操作子群114には、通常のみキシングコンソールと同様に各種の状態設定用の操作子が設けられている。そのうち、上述したのみキシング系統およびモニタ系統に関連する操作子の構成を図3を参照し説明する。

図において132はカスケード・オフ・スイッチであり、このスイッチが押下されるとエンジン間のカスケード接続が切断される（図5の一点鎖線で示された接続、および図6のカスケードケーブル290の接続）。なお、134はカスケード・マスタ・スイッチであり、このスイッチが押下されると、該コンソールの属するカスケードグループのエンジンがカスケードマスタに設定される。

【0087】

136はカスケード・スレーブ・スイッチであり、このスイッチが押下されると、当該コンソールの属するカスケードグループのエンジンがカスケードスレーブに設定される。上記スイッチ132、134、136は、何れのコンソールにおいても有効である。例えばデュアルコンソールのカスケード接続システムにおいては、コンソール100A～100Dの何れにおいてもカスケードモードを切り換えることができる。

【0088】

次に、138はトークバック・リンク・スイッチであり、カスケード接続された2台のコンソールシステムのトークバック信号のリンクのオン／オフ状態を切り換える。コンソール100Aにおけるトークバック・リンク・スイッチ138が操作されると、スイッチ360e、360fのオン／オフ状態が切り換えられる。また、コンソール100Bにおけるトークバック・リンク・スイッチ138が操作されると、スイッチ366e、366fのオン／オフ状態が切り換えられる。

【0089】

139はトークバックt o モニタBスイッチであり、一のコンソールに設けられたスイッチ139は、該一のコンソールのトークバック信号を、該一のコンソールとデュアルコンソールを構成する他のコンソールのモニタ信号MON_A（一のコンソールの側から見ればモニタ信号MON_B）にミキシングするか否かを指定する。例えば、コンソール100Aにおけるトークバックt o モニタBスイッチ139が操作されるとスイッチ324eのオン／オフ状態が切り換えられ、コンソール100Bにおける同スイッチ139が操作されると、スイッチ304eのオン／オフ状態が切り換えられる。

【 0 0 9 0 】

次に、140はコムイン・リンク・スイッチであり、コンソール100A～100Dの該スイッチが押下される毎に、スイッチ308e, 328e, 308f, 328fのオン／オフ状態が各々切り換えられる。すなわち、コンソール100Aにおけるコムイン・リンク・スイッチ140が操作されると、スイッチ308e, 308fのオン／オフ状態が切り換えられ、コンソール100Bにおけるスイッチ140が操作されると、スイッチ328e, 328fのオン／オフ状態が切り換えられる。

【 0 0 9 1 】

142, 143はカスケード・トークバックtoコムイン・スイッチであり、相手側のカスケードグループのコンソールからのトークバック信号を、当該スイッチ142, 143が設けられている一のコンソールのコムイン信号にリンクさせるか否かを切り換えるスイッチである。例えば、コンソール100Aにおいてスイッチ142がオン状態に設定されると、スイッチ322eがオン状態に設定されるとともに、これに連動して322fがオン状態に設定され、コンソール100A, 100C間の通話が可能になる。

【 0 0 9 2 】

また、コンソール100Aにおいてスイッチ143がオン状態に設定されると、スイッチ320eがオン状態に設定されるとともに、これに連動してスイッチ320fがオン状態に設定され、コンソール100A, 100D間の通話が可能になる。同様に、コンソール100Bにおけるスイッチ142, 143が操作されると、各々スイッチ342e, 340eのオン／オフ状態が切り換えられ、それと連動してスイッチ342f, 340fのオン／オフ状態も切り換えられる。

【 0 0 9 3 】

次に、144はVCAリンク・スイッチであり、このスイッチが押下される毎に、カスケードグループ間のVCAのリンクのオン／オフ状態が切り換えられる。ここで、VCAについて若干説明しておく。まず、ミキシングシステムにおける複数の入力チャンネルには各々フェーダが割り当てられているから、これらフェーダを操作することによってこれら複数の入力チャンネルの音量レベルを自在

に設定することができる。しかし、これら入力チャンネルが相互に関係を有する信号である場合は、1個のフェーダを操作することによってこれら入力チャンネルの全ての音量レベルを連動して調節できれば便利である。

【0094】

そこで、複数の各入力チャンネルに対応するフェーダに加えて、これら入力チャンネルの音量レベルを連動して増減する共通のフェーダが設けられることがある。かかる動作をVCAと呼び、複数の入力チャンネルに割り当てられる共通のフェーダをVCAフェーダと呼ぶ。VCAの設定内容としては、各VCAフェーダの有効／無効および、各VCAフェーダに対する入力チャンネルの割り当て状態である。VCAがリンクされると、かかる設定内容が両カスケードグループにおいて共通化される。

【0095】

次に、146はキュー・リンク・スイッチであり、相手側のカスケードグループ内の対応するコンソールとの間でキューリンクを行うか否かを設定するスイッチである。上述したデュアルコンソールをカスケード接続したシステムにおいては、コンソール100A、100Cのキュー・リンク・スイッチ146によってスイッチ274e、274f（図5参照）のオン／オフ状態が連動して切り換えられ、コンソール100B、100Dのキュー・リンク・スイッチ146によって、スイッチ280e、280fのオン／オフ状態が連動して切り換えられる。

【0096】

148はシーン・リンク・スイッチであり、カスケードグループ間においてシーンリコールを連動させるか否かを切り換えるスイッチである。なお、シーン・リンク・スイッチ148は各コンソール100A～100Dの何れにおいても有効である。また、149はキュー・リンク・スイッチであり、デュアルコンソールシステムにおける2つのコンソール間においてキューの操作を連動させるか否かを切り換えるスイッチである。なお、該スイッチ149は、マスタ／スレーブコンソールの何れにおいても有効である。

【0097】

3. 実施形態の動作

3. 1. カスケード接続に関する動作

3. 1. 1. タイマ割込み処理

各エンジンに接続されているコンソール（デュアルコンソールシステムにおいてはマスタコンソール）において、該エンジンがカスケードマスタまたはカスケードスレーブに設定されている場合には、所定時間毎に、図9に示すタイマ割込み処理ルーチンがCPU118において起動される。

図において処理がステップSP202に進むと、エンジンのカスケードI/O部206を介して他のエンジンが接続されているか否かが検出される。次に、処理がステップSP204に進むと、RAM122に記憶された「カスケード接続フラグ」が“1”であるか否かが判定される。なお、カスケード接続フラグはエンジン200の接続時において“0”にリセットされ、その後に当該エンジンに他のエンジンがカスケード接続された時に“1”に設定されるフラグである。

【0098】

カスケード接続フラグが“0”であった場合にはステップSP204においては「NO」と判定され、処理はステップSP210に進む。ここでは、他のエンジンがカスケードI/O部206を介して物理的に接続されているか否かが判定される。ここで「YES」と判定されると、処理はステップSP212に進み、相手側のエンジンの機種、バージョン、および設定状態が確認される。ここで、バージョンとは、フラッシュメモリ220に記憶されているファームウェアのバージョンであり、設定状態とは「カスケードマスタ」、「カスケードスレーブ」、または「カスケードオフ」の状態のことである。

【0099】

例えば、自機側のエンジンがカスケードマスタに設定されていれば、相手側は必ずカスケードスレーブでなければならず、自機側のエンジンがカスケードスレーブに設定されていれば、相手側は必ずカスケードマスタでなければならない。次に、処理がステップSP214に進むと、ステップSP212における確認結果に基づいて、自機側および相手側のエンジンがカスケード接続に適合しているか否かが判定される。すなわち、カスケード接続を行うためには、両エンジンの機種が同一であって、両者のファームウェアのバージョンも一致していなければ

ならず、両エンジンのうち一方がカスケードマスタ、他方がカスケードスレーブに設定されていなければならない。

【 0 1 0 0 】

確認結果がこの条件に適合していれば「YES」と判定され、処理はステップSP216に進む。ここでは、両エンジンの接続開始処理が実行される。具体的には、まずリンクされているパラメータ（例えばVCAの設定等）がカスケードマスタ側のコンソールからカスケードスレーブ側のコンソールに対してコピーされる。次に、ステップSP216においては、ミキシング系統およびモニタ系統のアルゴリズムが変更される。その詳細をデュアルコンソールのカスケード接続システム（図2(d)）の場合を例として説明しておく。

【 0 1 0 1 】

まず、ステップSP216の実行以前においては、エンジン200E, 200Fにおいて、各々独立したミキシング系統のアルゴリズム（図4参照）が構築されていた。これに対して、ミキシングバス、キューバス周辺のアルゴリズムが図5に示すように変更される。すなわち、ミキシングバス244e, 244fが相互にリンクされるとともに、スイッチ274e, 274fおよびスイッチ280e, 280fのオン／オフ状態に基づいてキューバス246e, 246fあるいはキューバス248e, 248fもリンクおよび解除が可能になる。

【 0 1 0 2 】

また、モニタ系統に関し、ステップSP216の実行以前においては、各エンジンにおいて図6、図7に示すモニタ系統のアルゴリズムが形成されていたが、カスケードグループ相互間の信号は存在しないものと看做されていた。換言すれば、カスケードケーブル290を通過する信号レベルは全て「0」とであると看做されていた。しかし、ステップSP216が実行されることにより、モニタ系統の信号が相互にやりとりされ、各コンソールにおいては相手側のカスケードグループにおけるトークバック信号等をコムイン信号等にミキシングすることなどが可能になる。

【 0 1 0 3 】

但し、本ルーチンにおいて実行される処理は、あくまでも自機側のエンジンの

アルゴリズムを設定する処理である。本ルーチンがコンソール 1 0 0 A において実行されたのであれば、エンジン 2 0 0 E のアルゴリズムのみが設定されるのである。一方、他方のカスケードグループにおけるコンソール 1 0 0 C においても同一のルーチンが実行されるため、これによってエンジン 2 0 0 F 側のアルゴリズムが設定される。このように両マスタコンソールにおいてステップ S P 2 1 6 の処理が完了することによって、両エンジン 2 0 0 E, 2 0 0 F におけるアルゴリズムの再構築が完了するのである。以上のように、ステップ S P 2 1 6 の処理が完了すると、次に処理はステップ S P 2 1 8 に進み、カスケード接続フラグが“1”に設定される。

【 0 1 0 4 】

なお、上述したステップ S P 2 1 0 において「N O」と判定されると、実質的な処理が行われることなく該タイマ割込み処理が終了する。また、ステップ S P 2 1 4 において「N O」と判定されると、処理はステップ S P 2 1 5 に進み、当該エンジンの表示器 2 1 4 において所定のエラー表示が行われる。このエラー表示においては、カスケード接続が成功しなかった旨と、その理由（機種の不一致、バージョンの不一致、または設定の矛盾）が表示される。さらに、このエンジンに接続されているコンソールに対してもエラーが発生した旨が通知され、当該コンソールの表示器 1 0 2 においても同様にエラーが表示される。

【 0 1 0 5 】

カスケード接続フラグが“1”に設定された後、再びタイマ割込み処理ルーチン（図 9）が起動されると、ステップ S P 2 0 2, S P 2 0 4 を介して処理はステップ S P 2 0 6 に進む。ここでは、カスケード接続の続行が不可能になったか否かが判定される。例えば、両エンジンを接続していたケーブルが外れた場合、あるいはエンジン 2 0 0 E, 2 0 0 F のカスケードモードが接続不能な状態（例えば双方がカスケードマスタ）に設定された場合がこれに該当する。

【 0 1 0 6 】

ステップ S P 2 0 6 において「Y E S」と判定されると、処理はステップ S P 2 0 8 に進み、接続停止処理が実行される。すなわち、ミキシング系統およびモニタ系統のアルゴリズムは、先にステップ S P 2 1 6 が実行される以前の状態に

戻る。次に、処理がステップ S P 2 0 9 に進むと、カスケード接続フラグが “ 0 ” に設定され、本ルーチンの処理が終了する。

【 0 1 0 7 】

3. 1. 2. シーンリコール処理

何れかのコンソールにおいてシーンリコール操作が実行されると、コンソールにおいて、図 1 0 (a) に示すシーンリコールイベント処理ルーチンが起動される。なお、ここでは主としてシングルコンソールシステムにおける動作について説明し、デュアルコンソールシステムにおける動作については後述する。

【 0 1 0 8 】

図において処理がステップ S P 2 3 0 に進むと、リコールされたシーンのシーン番号が変数 S N に代入される。次に、処理がステップ S P 2 3 2 に進むと、当該コンソールに対応するエンジンは他のエンジンとカスケード接続され、かつ、該カスケード接続においてシーンリコール操作がリンクされているか否かが判定される。ここで「 N O 」と判定されると、処理はステップ S P 2 3 4 に進む。

【 0 1 0 9 】

ここでは、当該コンソールにおけるシーン領域 1 2 2 b の内容のうち当該シーン番号 S N に係る部分が新たなカレント操作データとしてカレント領域 1 2 2 a にコピーされる。次に、処理がステップ S P 2 3 6 に進むと、該カレント操作データに基づいて、対応するエンジンの信号処理部 2 0 2 のアルゴリズムのパラメータ等が再設定される。これにより、当該エンジン単独でシーン番号 S N の内容が再現され、本ルーチンの処理が完了する。

【 0 1 1 0 】

一方、ステップ S P 2 3 2 において「 Y E S 」と判定されると、処理はステップ S P 2 3 8 に進み、相手側のカスケードグループに属するコンソールに対して、当該シーン番号 S N とともにリコール要求が送信される。以下、デュアルコンソールのカスケード接続システムにおいて、コンソール 1 0 0 A にてシーンリコール操作が発生した場合を例として説明する。シーンリコール操作が発生すると、相手側のカスケードグループに属するコンソール 1 0 0 C, 1 0 0 D に対して、当該シーン番号 S N とリコール要求とが送信されることになる。

【 0 1 1 1 】

次に、処理がステップ S P 2 4 0 に進むと、コンソール 1 0 0 A 内においてシーン領域 1 2 2 b 内のシーン番号 S N の内容が新たなカレント操作データとしてカレント領域 1 2 2 a にコピーされる。次に処理がステップ S P 2 4 4 に進むと、相手側グループのコンソール 1 0 0 C, 1 0 0 D の双方から「リンク可能応答」を受信し、またはタイムアウトが発生したか（ステップ S P 2 4 0 の終了後に所定時間が経過したか）か否かが判定される。ここで「N O」と判定されると、ステップ S P 2 4 4 の処理が繰り返される。

【 0 1 1 2 】

一方、ステップ S P 2 3 8 においてコンソール 1 0 0 A からコンソール 1 0 0 C, 1 0 0 D に対してリコール要求が送信されると、コンソール 1 0 0 C, 1 0 0 D においては図 1 0 (b) に示すリコール要求受信イベント処理ルーチンが各々起動される。図において処理がステップ S P 2 7 0 に進むと、送信されたシーン番号が変数 S N に代入される。次に、処理がステップ S P 2 7 2 に進むと、コンソール 1 0 0 C および D 内において、各々のシーン番号 S N のシーンデータがカレント領域 1 2 2 a にコピーされる。

【 0 1 1 3 】

次に、処理がステップ S P 2 7 4 に進むと、カスケード接続の相手側である（シーンリコール操作が発生した）コンソール 1 0 0 A に対して、リコール可能応答が送信される。次に、処理がステップ S P 2 7 6 に進むと、リンクされたパラメータが相手側から受信されたか否かが判定される。ここで「N O」と判定されると、処理はステップ S P 2 8 0 に進み、相手側からリコール開始指令を受信し、またはタイムアウトが発生したか（ステップ S P 2 7 4 の終了後に所定時間が経過したか）否かが判定される。ここで「N O」と判定されると、処理はステップ S P 2 7 6 に戻る。

【 0 1 1 4 】

従って、コンソール 1 0 0 A からパラメータあるいはリコール開始指令が供給されるまで、コンソール 1 0 0 C, 1 0 0 D においてはステップ S P 2 7 6, S P 2 8 0 が繰り返し実行されることになる。一方、コンソール 1 0 0 C, 1 0 0

Dにおいて共に上記ステップSP274が実行され、双方のリコール可能応答がコンソール100Aに受信されると、図10(a)におけるステップSP244において「YES」と判定され、処理はステップSP246に進む。

【0115】

ステップSP246においては、リンクされたパラメータが存在するか否かが判定される。ここで「YES」と判定されると、処理はステップSP248に進み、リンクされたパラメータがコンソール100C、100Dに送信される。なお、ここでいう「パラメータ」とは、シーン番号SNに属するパラメータである。例えば、両カスケードグループにおいて「VCA」がリンクされ、何れかのカスケードグループにおいて、このシーン番号SNに係るVCAの状態が変更されていたと仮定する。

【0116】

かかる場合には、当該シーンリコール操作が発生したコンソール100Aから、コンソール100C、100Dに対して、当該VCAに係る設定データが転送されるのである。さて、リンクされたパラメータがコンソール100Cまたは100Dにおいて受信されると、当該コンソールにおいては、パラメータが受信される毎にステップSP276において「YES」と判定され、ステップSP278が実行される。すなわち、受信したパラメータに応じてカレント操作データが逐次更新されるのである。

【0117】

以上のように、何れかのコンソールにおいてシーンリコール操作が生じると、リンクされたパラメータは「操作が発生したコンソール」から「他のコンソール」に送信される点に本実施形態の特徴の一つがある。すなわち、上述したタイマ割込み処理ルーチン（図9）においては、各種パラメータは必ず「カスケードマスタ側のコンソール」から「カスケードスレーブ側のコンソール」に送信されていたが、一旦カスケード接続が確立した後においては、リンクされたパラメータはカスケードマスタおよびカスケードスレーブの何れの側のコンソールにおいても編集することができる。これにより、何れのコンソールにおけるオペレータも、シーンリコール操作を行うことによって自機側のコンソールのリンクパラメー

タの設定内容を他のコンソールに反映させることができる。

【 0 1 1 8 】

さて、コンソール 1 0 0 A においては、リンクされた全てのパラメータが送信された後、処理はステップ S P 2 5 0 に進む。ここでは、コンソール 1 0 0 C, 1 0 0 D に対して、リコール開始指令が送信される。次に、処理がステップ S P 2 5 2 に進むと、カレント領域 1 2 2 a の内容に合致するように、エンジン 2 0 0 E の信号処理部 2 0 2 のアルゴリズムのパラメータ等が制御される。これにより、シーンリコール操作が発生したコンソール 1 0 0 A における処理が終了する。

【 0 1 1 9 】

一方、コンソール 1 0 0 C, 1 0 0 D においては、該リコール開始指令が受信されると、ステップ S P 2 8 0 において「 Y E S 」と判定され処理はステップ S P 2 8 2 に進む。ここでは、コンソール 1 0 0 C または 1 0 0 D のカレント領域 1 2 2 a の内容に合致するように、エンジン 2 0 0 F の信号処理部 2 0 2 のアルゴリズムのパラメータ等が制御される。

【 0 1 2 0 】

このように、本実施形態においては、カスケード接続時であってシーンがリンクされている場合に何れかのコンソールにおいてシーンリコール操作が発生すると、関係する全てのエンジンにおいてほぼ同時にシーンリコール操作が反映される（ステップ S P 2 5 2, 2 8 2）。これにより、例えばリコール要求を受信した側のコンソールまたはエンジンにおいて他の中断不可能な処理が実行されている場合であっても、各コンソールおよびエンジン毎にシーンリコールを行うタイミングがずれるような不具合を未然に防止することができる。

【 0 1 2 1 】

但し、ステップ S P 2 4 4 または 2 8 0 においてはタイムアウトの判定も行われるから、例えばリコール要求を送信または受信した側のコンソールにおいて比較的長い時間に渡って応答を行うことができない場合には、他方のコンソールにおいては独自にシーンを切り換えることが可能である。

【 0 1 2 2 】

3. 2. デュアルコンソールに関する動作

3. 2. 1. コンソールにおけるタイマ割込み処理

各コンソールは、オペレータによって、「デュアルコンソールオフ」、「デュアルコンソールマスタ」、または「デュアルコンソールスレーブ」のうち何れかの動作モードに設定される。これらの動作モードは、各々「シングルコンソールシステムのマスタコンソール」、「デュアルコンソールシステムのマスタコンソール」および「デュアルコンソールシステムのスレーブコンソール」の動作状態に対応している。換言すれば、各コンソールに対して希望する動作状態に応じて、オペレータが各々の動作モードを設定することになる。

【 0 1 2 3 】

ここで、動作モードとして「デュアルコンソールオフ」が選択された場合は、当該コンソールの動作状態は必ず「シングルコンソールシステムのマスタコンソール」に設定される。但し、動作モードとしてデュアルコンソールシステムのマスタコンソールあるいはスレーブコンソールが選択された場合、実際のコンソールの動作状態は該コンソールの動作モードと、実際の接続状態とに応じて決定される。

【 0 1 2 4 】

このため、動作モードが「デュアルコンソールマスタ」または「デュアルコンソールスレーブ」に設定されている場合には、各コンソールにおいては、所定時間毎に図 1 1 に示すタイマ割込みルーチンが起動される。図において処理がステップ S P 1 0 2 に進むと、デュアル I / O 部 1 0 6 を介して他のコンソールが接続されているか否かが検出される。次に、処理がステップ S P 1 0 4 に進むと、R A M 1 2 2 内に記憶されたデュアル接続フラグが“1”であるか否かが判定される。なお、デュアル接続フラグはコンソールの電源投入時において“0”にリセットされ、その後に当該コンソールのデュアル I / O 部 1 0 6 を介して他のコンソールが接続された時に“1”に設定されるフラグである。

【 0 1 2 5 】

デュアル接続フラグが“0”であった場合にはステップ S P 1 0 4 においては「N O」と判定され、処理はステップ S P 1 1 0 に進む。ここでは、他のコンソ

ールがデュアル I / O 部 1 0 6 を介して物理的に接続されているか否かが判定される。ここで「Y E S」と判定されると、処理はステップ S P 1 1 2 に進み、相手側のコンソールの機種、バージョン、および動作モードの設定状態が確認される。ここで、バージョンとは、フラッシュメモリ 1 2 0 に記憶されているファームウェアのバージョンである。

【 0 1 2 6 】

なお、このデュアル接続フラグは、デュアルコンソールシステムにおける各コンソールの動作状態を確定するフラグである。すなわち、本ルーチンにおいては、動作モードがデュアルコンソールマスタであったとしても、デュアルコンソールスレーブであったとしても、最初は当該コンソールはマスタコンソールであるものと仮定して各種処理が実行される。そして、デュアル接続フラグが“1”に設定されると、動作モードがデュアルコンソールマスタであるコンソールの動作状態はマスタコンソールに、動作モードがデュアルコンソールスレーブであるコンソールの動作状態はスレーブコンソールに、各々確定されることになる。

【 0 1 2 7 】

次に、処理がステップ S P 1 1 4 に進むと、ステップ S P 1 1 2 における確認結果に基づいて、自機および相手側がデュアルコンソールシステムに適合しているか否かが判定される。すなわち、両コンソールの機種が同一であって、かつ両コンソールのファームウェアのバージョンも一致していなければならない。さらに、自機の動作モードがデュアルコンソールマスタであれば、相手側の動作モードは必ずデュアルコンソールスレーブでなければならず、自機がデュアルコンソールスレーブであれば、相手側は必ずデュアルコンソールマスタでなければならない。

【 0 1 2 8 】

確認結果がこの条件に適合していれば「Y E S」と判定され、処理はステップ S P 1 1 6 に進む。ここでは、当該コンソールの動作モードがデュアルコンソールマスタに設定されているか否かが判定される。

【 0 1 2 9 】

ここで「Y E S」と判定されると、処理はステップ S P 1 1 7 に進み、デュア

ルコンソールスレーブに設定された相手側のコンソールとの間で、カレント操作データ、シーンデータおよびライブラリデータが比較される。なお、比較にあたっては、これらデータを全て転送すると膨大な転送時間が必要になるため、スレーブコンソールからチェックサム結果およびタイムスタンプを受信し、これらに基づいて比較される。

【 0 1 3 0 】

次に、処理がステップ S P 1 1 8 に進むと、ステップ S P 1 1 6 の比較結果において不整合が存在するか否かが判定される。不整合が存在した場合には「Y E S」と判定され、処理はステップ S P 1 2 0 に進み、不整合に係るデータを整合させるか否かをオペレータに問い合わせるポップアップウィンドウが表示器 1 0 2 に表示される。このポップアップウィンドウにおいては、「一致しないデータを相手側コンソールに転送しますか？」というメッセージと、転送予想時間（例えば「2 0 分」）と、「OK」ボタンと、「キャンセル」ボタンとが表示される。

【 0 1 3 1 】

ところで、マスタコンソールからスレーブコンソールに対して転送し得るデータは、カレント操作データ、シーンデータおよびライブラリデータの 3 種類であり、上記ポップアップウィンドウはこれらデータのうち不整合が生じたデータ毎に表示される。すなわち、ポップアップウィンドウは最大で合計 3 回表示される。何れかのウィンドウにおいてオペレータが「OK」ボタンをマウスでクリックすると、マスタコンソールからスレーブコンソールに対して、対応するデータが転送され、スレーブコンソール内の対応する領域 1 2 2 a, 1 2 2 b または 1 2 2 c 内に該データが順次転送されてゆく。なお、シーン領域 1 2 2 b には、最大約「1 0 0 0」セットのシーンデータが記憶されるが、これらに不整合が存在するか否かはシーンデータ毎に判断されるため、一致しないシーンデータの数が少ないければ転送時間も短くなる。

【 0 1 3 2 】

また、オペレータは、転送途中に「キャンセル」ボタンをマウスでクリックすることにより、当該転送を随時停止させることができる。3 種類のデータの全て

について転送が完了し、あるいは「キャンセル」ボタンが押下された場合は、処理はステップ S P 1 2 2 に進む。換言すれば、マスタコンソールおよびスレーブコンソール間においてシーンデータ等を完全に一致させなくても、これらをデュアルコンソールシステムとして動作されることが可能である。例えばシーンの切換等を行わないのであれば、両コンソールのシーンデータを異なる状態のまま残しておいても差し支え無い。かかる機能は、特にデュアルコンソールシステムを迅速に立ち上げる必要がある場合に用いて好適である。

【 0 1 3 3 】

次に、処理がステップ S P 1 2 2 に進むと、二台のコンソール間において接続開始処理が行われる。すなわち、後述する操作イベント処理ルーチン等（図 1 3 (a)～(d)）が有効にされ、一方のコンソールにおける操作が他方のコンソールにおいても反映されるようになる。次に、処理がステップ S P 1 2 3 に進むと、デュアル接続フラグが“1”に設定される。以上のステップが終了すると、処理はステップ S P 1 2 4（図 1 2）に進む。

【 0 1 3 4 】

なお、上述したステップ S P 1 1 0 において「NO」と判定されると、上記ステップ S P 1 1 2 ～ S P 1 2 3 はスキップされ、処理は直ちにステップ S P 1 2 4 に進む。また、ステップ S P 1 1 4 において「NO」と判定されると、処理はステップ S P 1 1 5 に進み、当該コンソールの表示器 1 0 2 において所定のエラー表示が行われ、しかる後に処理はステップ S P 1 2 4 に進む。なお、このエラー表示においては、デュアルコンソールシステムの構築が成功しなかった旨と、その理由（機種の不一致、バージョンの不一致、または設定の矛盾）が表示される。

【 0 1 3 5 】

また、当該ルーチンを実行しているコンソールの動作モードがデュアルコンソールスレーブに設定されていた場合は、ステップ S P 1 1 6 において「NO」と判定され、処理は直ちにステップ S P 1 2 2 に進む。これにより、スレーブコンソールにおいては上述したようなポップアップウィンドウ等が表示されないままマスタコンソールとの間の接続開始処理が実行される。

【 0 1 3 6 】

さて、ステップ S P 1 2 4（図 1 2）においては、当該コンソールがスレーブコンソールとして確定しているか否かが判定される。上述したように、動作モードがデュアルコンソールスレーブであって、かつ、デュアル接続フラグが“1”であれば当該コンソールはスレーブコンソールに確定する。かかる場合は、エンジン接続に係るステップ S P 1 2 5～S P 1 3 8 の処理はスキップされる。換言すれば、スレーブコンソールに確定したコンソールに対して仮にエンジンが接続されていたとしても、そのエンジンに対してなんら処理が行われなくなる。

【 0 1 3 7 】

当該コンソールがスレーブコンソールとして確定していなければ処理はステップ S P 1 2 5 に進む。動作モードがデュアルコンソールスレーブに設定され未だデュアル接続フラグが“0”であるコンソールもこれに該当するため、処理はステップ S P 1 2 5 に進む。同ステップにおいては、エンジン接続フラグ“1”であるか否かが判定される。ここで該フラグが“0”であれば「NO」と判定され、処理はステップ S P 1 3 0 に進む。ここでは、エンジンがデータ I / O 部 1 1 0 および通信 I / O 部 1 1 2 を介して物理的に接続されているか否かが判定される。ここで「YES」と判定されると、処理はステップ S P 1 3 2 に進み、当該エンジンの機種、およびファームウェアのバージョンが確認される。

【 0 1 3 8 】

次に、処理がステップ S P 1 3 4 に進むと、ステップ S P 1 3 2 における確認結果に基づいて、当該コンソールに対して当該エンジンが適合しているか否かが判定される。エンジンがコンソールに適合していれば「YES」と判定され、処理はステップ S P 1 3 6 に進む。ここでは、カレント領域 1 2 2 a の内容に基づいて、当該エンジン内の信号処理部 2 0 2 の状態が設定される。

【 0 1 3 9 】

次に、処理がステップ S P 1 3 8 に進むと、エンジン接続フラグが“1”に設定され、本ルーチンの処理が終了する。なお、上述したステップ S P 1 3 0 において「NO」と判定されると、上記ステップ S P 1 3 2 ～S P 1 3 8 はスキップ

され、本ルーチンの処理は直ちに終了する。また、ステップ S P 1 3 4 において「N O」と判定されると、処理はステップ S P 1 3 5 に進み、当該コンソールの表示器 1 0 2 において所定のエラー表示が行われた後に本ルーチンの処理が終了する。なお、このエラー表示においては、エンジンへの接続が成功しなかった旨と、その理由（機種、バージョンの不適合など）が表示される。

【 0 1 4 0 】

以上の処理により、「マスタコンソール」および「スレーブコンソール」の区別が確定される。すなわち、デュアル接続フラグおよびエンジン接続フラグが共に“1”であるコンソールは「マスタコンソール」であり、デュアル接続フラグが“1”であってエンジン接続フラグが“0”であるコンソールは「スレーブコンソール」である。

【 0 1 4 1 】

ところで、デュアル接続フラグが“1”に設定された後、再びタイマ割込み処理ルーチン（図 1 1）が起動されると、ステップ S P 1 0 2， S P 1 0 4 を介して処理はステップ S P 1 0 6 に進む。ここでは、デュアルコンソールシステムの続行が不可能になったか否かが判定される。例えば、両コンソールを接続していたケーブルが外れた場合、あるいは両コンソールが共にマスタコンソールに設定された場合がこれに該当する。ステップ S P 1 0 6 において「Y E S」と判定されると、ステップ S P 1 0 8 において接続停止処理が実行される。次に、処理がステップ S P 1 0 9 に進むと、デュアル接続フラグ“0”に設定され、ステップ S P 1 2 5 以下の処理が実行されることになる。

【 0 1 4 2 】

ステップ S P 1 0 8， S P 1 0 9 がこれまでのマスタコンソールにおいて実行された場合、およびこれまでのスレーブコンソールにおいて実行された場合の何れにおいても、該コンソールはシングルコンソールとして機能するようになる。

【 0 1 4 3 】

また、エンジン接続フラグが“1”に設定された後、再びタイマ割込み処理ルーチン（図 1 1）が起動されると、マスタコンソールにおいてはステップ S P 1 2 5 を介して処理はステップ S P 1 2 6 に進む。ここでは、エンジンに対する接

続が切断されたか否かが判定される。例えば、コンソールとエンジン間のケーブルが外れた場合、あるいはエンジンの電源がオフになった場合等がこれに該当する。ステップ S P 1 2 6 において「Y E S」と判定されると、ステップ S P 1 2 8 において接続停止処理が実行され、ステップ S P 1 2 9 においてエンジン接続フラグが“0”に設定される。

【 0 1 4 4 】

3. 2. 2. マスタコンソールタイマ割込み処理：図 1 3 (d)

マスタコンソール（またはシングルコンソール）においては、所定時間毎に図 1 3 (d) に示すタイマ割込み処理ルーチンが起動される。なお、本ルーチンは、図 1 1 におけるタイマ割込みルーチンよりも高頻度に行われる。図 1 3 (d) において処理がステップ S P 1 8 0 に進むと、カレント操作データに変化が生じたか否かが判定される。カレント操作データは、次に説明する操作イベント処理ルーチン（図 1 3 (a)）によって更新される。ここで「Y E S」と判定されると、処理はステップ S P 1 8 2 に進み、変化した後のデータに基づいて、対応するエンジン内のミキシングシステムのアルゴリズムのパラメータ等が更新される。このルーチンにより、マスタコンソール（またはシングルコンソール）のカレント操作データに基づいて、ミキシング処理の内容が制御される。

【 0 1 4 5 】

3. 2. 3. 操作イベント処理ルーチン：図 1 3 (a)

マスタ／スレーブを問わず、何れかのコンソールの電動フェーダ部 1 0 4 あるいは操作子群 1 1 4 において、所定の操作イベントが発生すると、図 1 3 (a) に示す操作イベント処理ルーチンが起動される。ここで、「所定の操作イベント」とは、ミキシングシステムに対して変化を与える操作であり、シーンリコール操作、電動フェーダの操作、音質調整用の操作等がこれに含まれる。従って、キュー信号 C U E やモニタ信号 M O N _ A に関する設定、操作子の割当設定（どの操作子にどのような機能を割り当てるのかの設定）等の操作は該「所定の操作イベント」には含まれない。

【 0 1 4 6 】

図において処理がステップ S P 1 5 0 に進むと、操作されたパラメータを識別

するパラメータ番号が変数 P N に、また、当該パラメータについて操作後の新しい値が変数 B U F に代入される。次に、処理がステップ S P 1 5 2 に進むと、操作が発生した当該コンソールが他のコンソールとの間で接続されデュアルコンソールシステムを構成しているか否かが判定される。

【 0 1 4 7 】

ここで「Y E S」と判定されると、処理はステップ S P 1 5 4 に進み、発生した操作イベントの内容、すなわちパラメータ番号 P N とパラメータ値 B U F とがデュアル I / O 部 1 0 6 を介して相手側のコンソールに送信される。なお、当該コンソールがシングルコンソールシステムを構成する場合にはステップ S P 1 5 2 において「N O」と判定され、ステップ S P 1 5 4 は実行されない。次に、処理がステップ S P 1 5 6 に進むと、操作内容に応じて、カレント領域 1 2 2 a 内のカレント操作データが更新される。発生した操作イベントが電動フェーダの操作であった場合は、ステップ S P 1 5 6 において、該電動フェーダの位置に応じて、カレント操作データ中の該電動フェーダに割り当てられた入力チャンネルないし出力チャンネルの音量を制御するデータが更新される。また、発生した操作イベントがシーンリコールの操作であった場合は、ステップ S P 1 5 6 において、上述したシーンリコールイベント処理ルーチン（図 1 0 (a)）が呼び出される。

【 0 1 4 8 】

ところで、デュアルコンソールシステムでシーンリコールの操作イベントが発生した場合は、パラメータ番号 P N は「シーンリコール」を示す値に設定され、パラメータ値 B U F はシーン番号に設定される。ここで、マスタ／スレーブコンソールにおいて同一のシーン番号を有するシーンデータが異なっている可能性もあるが、本ルーチンにおいてはこれらシーンデータの異同は考慮されない。ここに本実施形態の特徴の一つがある。すなわち、本実施形態においては、シーンリコール時にコンソール間でやりとりされる情報はパラメータ番号 P N とパラメータ値 B U F のみであり、伝送される情報量をきわめて小さくすることができる。これにより、両コンソールは、自機が有するシーンデータに基づいて、迅速にシーンの切換を実行することができる。

【 0 1 4 9 】

3. 2. 4. 操作イベント受信処理ルーチン：図 1 3 (b)

上記ステップ S P 1 5 4 において、操作が発生したコンソールから操作イベント内容が送信されると、この操作イベント内容を受信した側のコンソールにおいては図 1 3 (b)に示す操作イベント受信処理ルーチンが起動される。

【 0 1 5 0 】

図において処理がステップ S P 1 6 0 に進むと、受信したパラメータ番号およびパラメータ値が各々変数 P N , B U F に代入される。次に、処理がステップ S P 1 6 2 に進むと、カレント操作データに対して、該パラメータ番号 P N およびパラメータ値 B U F が整合性を有するか否かがチェックされる。

【 0 1 5 1 】

すなわち、デュアルコンソールシステムにおいては両コンソールのカレント操作データは一致していることが望ましいが、先にステップ S P 1 2 0 において説明したように、両コンソールのカレント操作データまたはシーンデータに相違があったとしても、これを見捨ててデュアルコンソール動作を開始させることができる。カレント操作データの相違を見捨てた場合には、両コンソールには当初から不整合が生じる可能性がある。また、何れかのシーンデータに不整合があった場合には、両コンソールにおいて当該シーンデータがリコールされた時にカレント操作データに不整合が生じる場合がある。

【 0 1 5 2 】

ここで、「不整合」の意味について若干説明しておく。「不整合」は、「あるパラメータ（入力チャンネルのペア設定、エフェクトの選択など）の設定により、パラメータ数が増減したり他のパラメータの機能が変更される」場合等に発生する。例えば、「パラメータ番号で指定されたパラメータが有効ではない場合」、あるいは「パラメータ番号で指定されたパラメータに、その変化許容範囲から外れるようなパラメータ値を設定しようとした場合」などが挙げられる。

【 0 1 5 3 】

次に、処理がステップ S P 1 6 4 に進むと、該 S P 1 6 2 のチェック結果に基づいて、操作イベントの整合性が有ったか否かが判定される。整合性があれば「

YES」と判定され、処理はステップSP166に進み、受信した操作イベントに応じてカレント操作データが更新される。また、ステップSP164において「NO」と判定されると、処理はステップSP168に進み、スレーブコンソール側の表示器102において、不整合が生じた旨の警告表示が行われる。以上のステップにより、本ルーチンの処理が終了する。

【0154】

該ステップSP168の処理は、実際には、このルーチンがマスタコンソールで実行されるのかスレーブコンソールで実行されるのかによって異なる。すなわち、マスタコンソールにおいてステップSP168が実行される場合には、マスタコンソールからスレーブコンソールに対して、警告表示を行うようにコマンドが出力され、これがスレーブコンソールにおいて受信されると、スレーブコンソールにおいて警告表示が行われる。また、スレーブコンソール側においてステップSP168が実行される場合には、単にスレーブコンソール側のCPU118の制御の下、スレーブコンソールの表示器102に警告表示が行われるだけである。

【0155】

以上の動作によれば、操作イベントに不整合が生じた場合の現象は、操作イベントが発生したコンソールに応じて異なることが解る。すなわち、元々当該操作イベントがマスタコンソールにおいて発生したのであれば、ステップSP156において当該操作イベントに基づいてマスタコンソール側のカレント操作データが更新される。そして、エンジン200は、マスタコンソール側のカレント操作データに基づいてアルゴリズムのパラメータ等を設定するため、操作内容がそのままパラメータに反映され、出力される音声信号に変化が生ずる。すなわち、マスタコンソール側から見れば、操作内容に応じて適切に音声信号に変化が生じることになる。

【0156】

一方、この不整合のある操作イベントがスレーブコンソールにおいて発生すると、スレーブコンソールにおいてステップSP156が実行される。しかし、スレーブコンソール側のカレント操作データは、エンジン200のアルゴリズムの

パラメータには反映されない。また、マスタコンソール側においてはステップ S P 1 6 4 において「NO」と判定され、ステップ S P 1 6 6 が実行されないから、マスタコンソール側のカレント操作データが更新されることはない。このため、スレーブコンソール側から見れば、該当する操作子をいくら操作しても全く音声信号に変化が見られないという現象が生じる。かかる理由により、上記ステップ S P 1 6 8 においては、スレーブコンソールにおいて警告表示を行うようにしたものである。

【 0 1 5 7 】

3. 2. 5. ベリファイ画面の表示

マスタコンソールにおいて所定の画面選択操作が行われると、図 1 4 に示すベリファイ／コピー画面が該コンソールの表示器 1 0 2 に表示される。図 1 4 において 4 0 2 はアップデートボタンであり、これがマウスでクリックされると、図 1 3 (c) に示すベリファイ開始イベント処理ルーチンが起動される。なお、このルーチンは、マスタおよびスレーブコンソールにおけるカレント操作データ、シーンデータおよびライブラリデータに相違があるか否かを確認するためのルーチンである。

【 0 1 5 8 】

図 1 3 (c) において処理がステップ S P 1 7 0 に進むと、変数 i に「0」が代入される。次に、処理がステップ S P 1 7 2 に進むと、第 i 番目のデータ（カレント操作データ、シーンデータまたはライブラリデータ）のチェックサムおよびタイムスタンプを送信するようにスレーブコンソールに対して要求される。これに応答してスレーブコンソールからチェックサムおよびタイムスタンプが供給されると、処理はステップ S P 1 7 4 に進む。ここでは、スレーブコンソールから供給されたチェックサムおよびタイムスタンプと、マスタコンソールに記憶されている第 i 番目のデータのチェックサムおよびタイムスタンプとが比較され、その比較結果が R A M 1 2 2 内の所定領域内に記録されとともに、該比較結果に基づいてベリファイ／コピー画面（図 1 4）の内容が更新される。

【 0 1 5 9 】

次に処理がステップ S P 1 7 4 に進むと、変数 i は最大値 i_MAX 未満であるか

否かが判定される。ここで「YES」と判定されると、ステップSP178において変数*i*が「1」だけインクリメントされる。以下、変数*i*が最大値*i*_MAXに達するまで、各データについてステップSP172, SP174の処理が繰り返される。ステップSP176において「NO」と判定され本ルーチンの処理が終了すると、ベリファイ／コピー画面（図14）は最新の情報に基づいて更新されていることになる。

【0160】

図14において、404は全体異同表示部であり、先のステップSP174の比較結果のうち少なくとも一のデータに相違があった場合には「DIFF」と表示され、全データが一致していた場合には「SAME」と表示される。次に、406はシーンデータ表示指令ボタンであり、このボタンがマウスでクリックされると、後述するライブラリ・リスト部430にシーンデータの詳細内容が表示される。408はシーンデータ異同表示部であり、いずれかのシーン番号についてシーンデータに相違があれば「DIFF」と表示され、全シーンデータが一致していれば「SAME」と表示される。なお、後述する他の異同表示部も、同様の表示方法によってデータの異同を表示する。

【0161】

410はライブラリデータ表示指令ボタン群であり、ユニットライブラリ、パッチライブラリ、ネームライブラリ等のライブラリデータ毎に設けられた複数の表示指令ボタンから構成されており、何れかのボタンがマウスでクリックされると、ライブラリ・リスト部430に対応するライブラリデータの詳細内容が表示される。412はライブラリデータ異同表示部群であり、各ライブラリデータ毎に、マスタ／スレーブコンソールの異同を表示する。

【0162】

420はカレント操作データ状態表示部であり、その内部に設けられたカレント異同表示部424には、マスタコンソール（図中では「CONSOLE 1」）およびスレーブコンソール（同「CONSOLE 2」）におけるカレント操作データの異同を表示する。422はコピー指示ボタンであり、マウスでクリックされるとマスタコンソールのカレント操作データがスレーブコンソールにコピーされる。

【 0 1 6 3 】

また、ライブラリ・リスト部 4 3 0 には、シーンデータ表示指令ボタン 4 0 6 あるいはライブラリデータ表示指令ボタン群 4 1 0 によって選択されたシーンデータまたはライブラリデータの詳細が表示される。なお、図示の例においては、シーンデータの詳細が表示されている。ライブラリ・リスト部 4 3 0 は複数の「列」から構成されており、ナンバ列 4 4 0 には、各データのナンバが表示される。4 4 2, 4 4 6 は項目名表示列であり、各データのデータ名称が表示される。4 4 8 は異同表示列であり、各データ毎の異同を表示する。

【 0 1 6 4 】

4 4 4 はコピー指示ボタン列であり、マウスでクリックされるとマスタコンソールの対応するデータがスレーブコンソールにコピーされる。また、ライブラリ・リスト部 4 3 0 は複数の行 4 3 6, 4 3 6, ……から構成されているが、最上位行 4 3 4 はシーンデータまたはライブラリデータ全体を表現している。すなわち、最上位行 4 3 4 における異同表示列 4 4 8 は、少なくとも一のデータに相違があれば「DIFF」と表示され、全データが一致する場合にのみ「SAME」と表示される。また、最上位行 4 3 4 におけるコピー指示ボタンがマウスでクリックされると、シーンデータまたは当該ライブラリデータのうち相違のあるデータ全体がマスタコンソールからスレーブコンソールにコピーされる。また、最上位行以外の行 4 3 6 におけるコピー指示ボタンがマウスでクリックされると、シーンデータまたはライブラリデータのうちその行に対応するデータがマスタコンソールからスレーブコンソールにコピーされる。4 5 0 はスクロールバーであり、最上位行 4 3 4 を除く行 4 3 6, 4 3 6, ……を上下方向にスクロールする。

【 0 1 6 5 】

なお、上述した操作イベント処理ルーチン（図 1 3 (a)）および操作イベント受信処理ルーチン（図 1 3 (b)）によれば、何れかのコンソールにおいてシーンリコールあるいはライブラリリコール操作があると、他方のコンソールにおいては、該シーンデータまたはライブラリデータに対するベリファイが自動的に実行される（S P 1 6 2）。従って、オペレータは、かかるリコール操作を行った後、ベリファイ／コピー画面（図 1 4）を表示器 1 0 2 に表示させると、特にアッ

ブデートボタン 4 0 2 を操作することなく、リコールしたシーンデータまたはライブラリデータの異同を確認することができる。

【 0 1 6 6 】

4. 変形例

本発明は上述した実施形態に限定されるものではなく、例えば以下のように種々の変形が可能である。

(1) 上記実施形態においては、コンソールまたはエンジン上で動作するプログラムによって各種処理を実行したが、このプログラムのみを C D - R O M 、フレキシブルディスク等の記録媒体に格納して頒布し、あるいは伝送路を通じて頒布することもできる。

(2) 上記実施形態においては、コンソールとエンジンを別体のものとして構成したが、これらは一体型にしてもよい。

(3) 上記実施形態においては、全てのモニタ系統、すなわち第 1 のモニタ系統（モニタ用セクタ 2 5 0 、第 1 モニタ信号 M O N 1 、コムイン信号 C O M M _ I N _ 1 ）、第 2 のモニタ系統（モニタ用セクタ 2 5 2 、第 2 モニタ信号 M O N 2 、コムイン信号 C O M M _ I N _ 2 ）、および第 1 キュー信号 C U E 1 （キューバス 2 4 6 ）、第 2 キュー信号 C U E 2 （キューバス 2 4 8 ）は、ステレオ構成とされる場合が多いが、モノラル構成であってもよいし、あるいは 5 . 1 チャンネル等の多チャンネル構成であってもよい。

(4) 上記実施形態においては、図 3 に示した各スイッチ 1 3 2 ~ 1 4 9 は各コンソールに 1 セットずつ設けられていたが、これらのスイッチを 2 セットずつ設け、デュアルコンソールシステムを構成する各コンソールにおいて相手側のコンソールの状態も制御できるようにしてもよい。

(5) 上記実施形態のステップ S P 2 1 6 においては、エンジン 2 0 0 E , 2 0 0 F において、各々独立したミキシングバス 2 4 4 e , 2 4 4 f が自動的にリンクされた（図 5 参照）。しかし、ミキシングバス 2 4 4 e , 2 4 4 f の全「 4 8 」本のバスを全てリンクする必要はなく、リンクのオン／オフ状態をバス毎に指定できるように、各バス毎にオン／オフスイッチを設けてもよい。

【 0 1 6 7 】

【発明の効果】

以上説明したように、この発明によれば、他のデジタルミキサから入力されたカスケード信号と、一のデジタルミキサにおいて生成され遅延された入力加算信号とを加算することによって、該カスケード信号の伝送遅延によって生じる位相差を入力加算信号において補償することができるから、各デジタルミキサにおいては位相の揃ったミキシング結果を得ることができ、相互にミキシング結果をやりとりしつつ高い独立性を確保することが可能である。

【図面の簡単な説明】

【図 1】 コンソール 1 0 0 およびエンジン 2 0 0 のハードウェアブロック図である。

【図 2】 本実施形態において構成可能な各種ミキシングシステムのブロック図である。

【図 3】 操作子群 1 1 4 の要部の外観図である。

【図 4】 1 台のエンジン 2 0 0 によって実現されるミキシングシステムアルゴリズムのブロック図である。

【図 5】 2 台のエンジン 2 0 0 E, 2 0 0 F によって実現されるカスケード接続システムにおけるミキシングシステムアルゴリズムの要部のブロック図である。

【図 6】 デュアルコンソールシステムのカスケード接続におけるモニタシステムのアルゴリズムのブロック図（1 / 2）である。

【図 7】 デュアルコンソールシステムのカスケード接続におけるモニタシステムのアルゴリズムのブロック図（2 / 2）である。

【図 8】 各コンソールの物理的配置例を示す図である。

【図 9】 マスタコンソールにおいて実行されるタイマ割込み処理ルーチンのフローチャートである。

【図 1 0】 シーンリコールイベント処理ルーチンおよびリコール要求受信イベント処理ルーチンのフローチャートである。

【図 1 1】 各コンソールにおいて実行される他のタイマ割込みルーチンのフローチャート（1 / 2）である。

【図 1 2】 各コンソールにおいて実行される他のタイマ割込みルーチンのフローチャート（2 / 2）である。

【図 1 3】 各種イベント処理ルーチンのフローチャートである。

【図 1 4】 表示器 1 0 2 に表示されるベリファイ / コピー画面を示す図である。

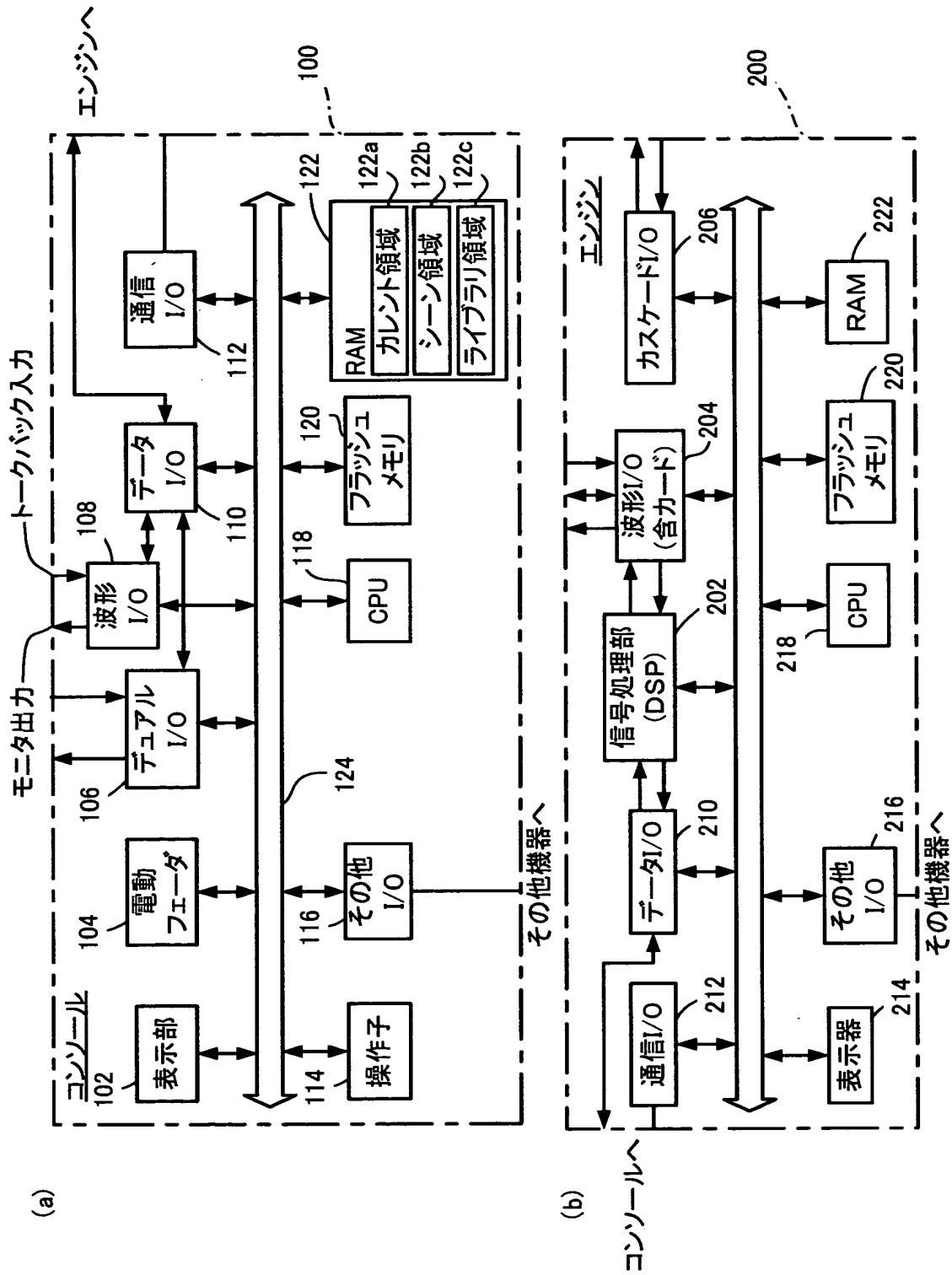
【符号の説明】

1 0 0, 1 0 0 A ~ 1 0 0 D … コンソール、1 0 2 … 表示器、1 0 4 … 電動フェーダ部、1 0 6 … デュアル I / O 部、1 0 8 … 波形 I / O 部、1 1 0 … データ I / O 部、1 1 2 … 通信 I / O 部、1 1 4 … 操作子群、1 1 6 … その他 I / O 部、1 1 8 … CPU、1 2 0 … フラッシュメモリ、1 2 2 … RAM、1 2 2 a … カレント領域、1 2 2 b … シーン領域、1 2 2 c … ライブラリ領域、1 2 4 … バス、1 3 2 … カスケード・オフ・スイッチ、1 3 4 … カスケード・マスタ・スイッチ、1 3 6 … カスケード・スレーブ・スイッチ、1 3 8 … トークバック・リンク・スイッチ、1 3 9 … トークバック t o モニタ B スwitch、1 4 0 … コムイン・リンク・スイッチ、1 4 2, 1 4 3 … カスケード・トークバック t o コムイン・スイッチ、1 4 4 … VCA リンク・スイッチ、1 4 6 … キュー・リンク・スイッチ、1 4 8 … シーン・リンク・スイッチ、1 4 9 … キュー・リンク・スイッチ、1 5 2 a, 1 5 2 b … モニタアンプ、1 5 4 a, 1 5 4 c … スwitch、2 0 0, 2 0 0 E, 2 0 0 F … エンジン、2 0 2 … 信号処理部、2 0 4 … 波形 I / O 部、2 0 6 … カスケード I / O 部、2 1 0 … データ I / O 部、2 1 2 … 通信 I / O 部、2 1 4 … 表示器、2 1 6 … その他 I / O 部、2 1 8 … CPU、2 2 0 … フラッシュメモリ、2 2 2 … RAM、2 2 4 … バス、2 3 2 … アナログ入力部、2 3 4 … デジタル入力部、2 3 6 … 内蔵エフェクタ、2 3 8 … 内蔵イコライザ、2 4 0 … 入力パッチ部、2 4 2 … 入力チャンネル調整部、2 4 4, 2 4 4 e, 2 4 4 f … ミキシングバス、2 4 6, 2 4 8, 2 4 6 e, 2 4 6 f, 2 4 8 e, 2 4 8 f … キューバス、2 5 0, 2 5 2 … モニタ用セレクタ、2 5 4 … 出力チャンネル調整部、2 5 4 e, 2 5 4 f … 出力チャンネル調整部、2 5 6 … マトリクス出力チャンネル部、2 5 7 … トークバックアウトスイッチ、2 5 8 … 出力パッチ部、2 6 0 … アナログ出力部、2 6 2 … デジタル出力部、2 6 4 e, 2 6 4 f … 遅延回

路、2 6 6 e, 2 6 6 f …加算器、2 7 0 e, 2 7 0 f …遅延回路、2 7 2 e, 2 7 2 f …加算器、2 7 4 e, 2 7 4 f …スイッチ、2 7 6 e, 2 7 6 f …遅延回路、2 7 8 e, 2 7 8 f …加算器、2 8 0 e, 2 8 0 f …スイッチ、2 9 0 …カスケードケーブル。

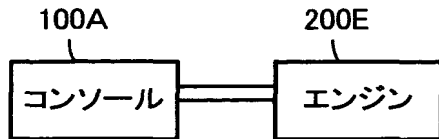
【書類名】 図面

【図 1】

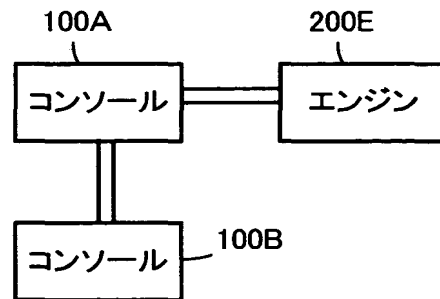


【図 2】

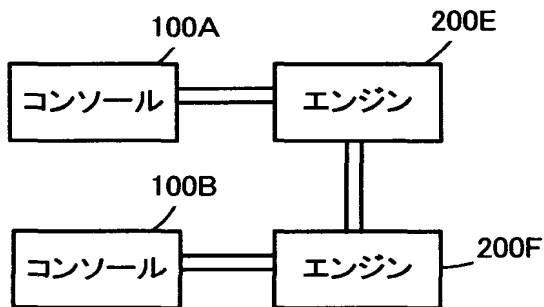
(a) シングルコンソール



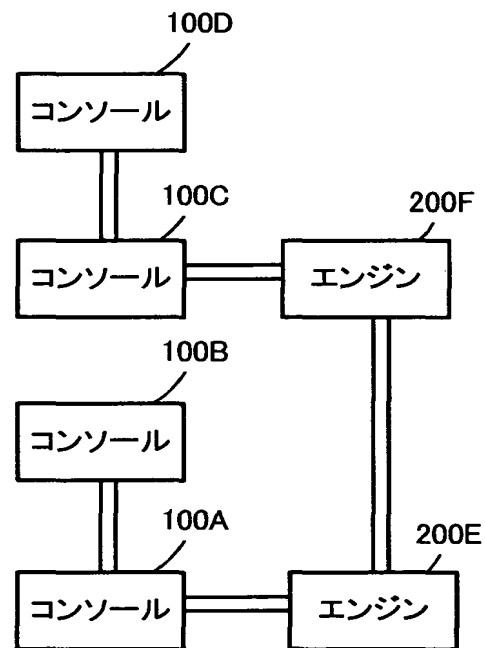
(b) デュアルコンソール



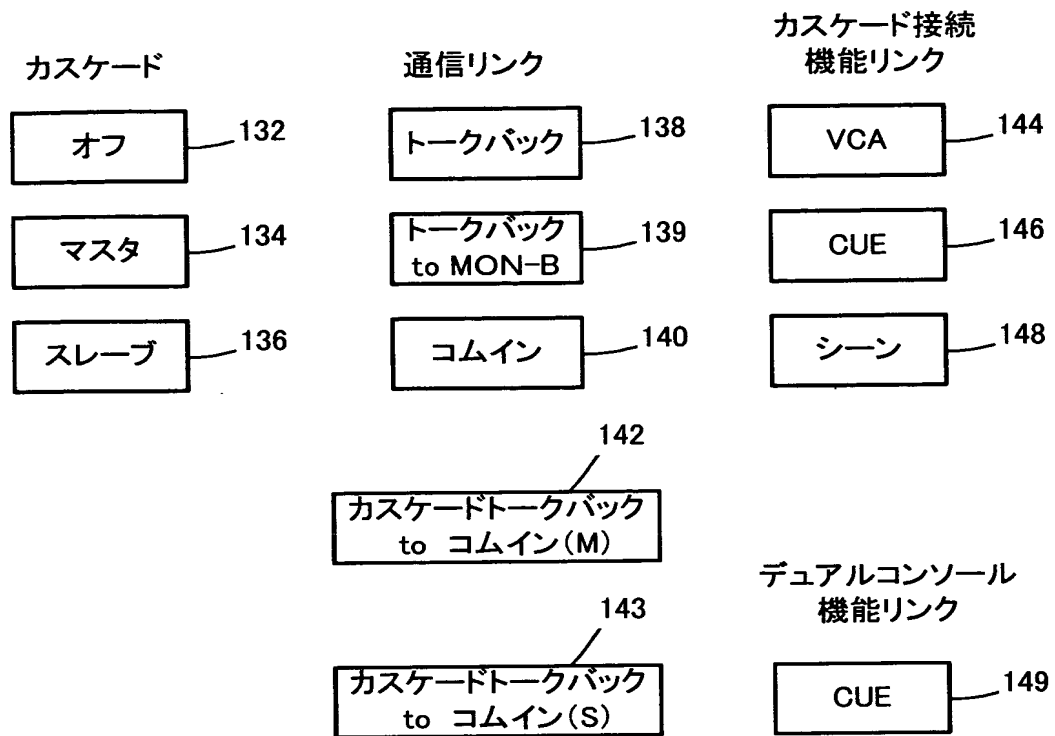
(c) カスケード接続



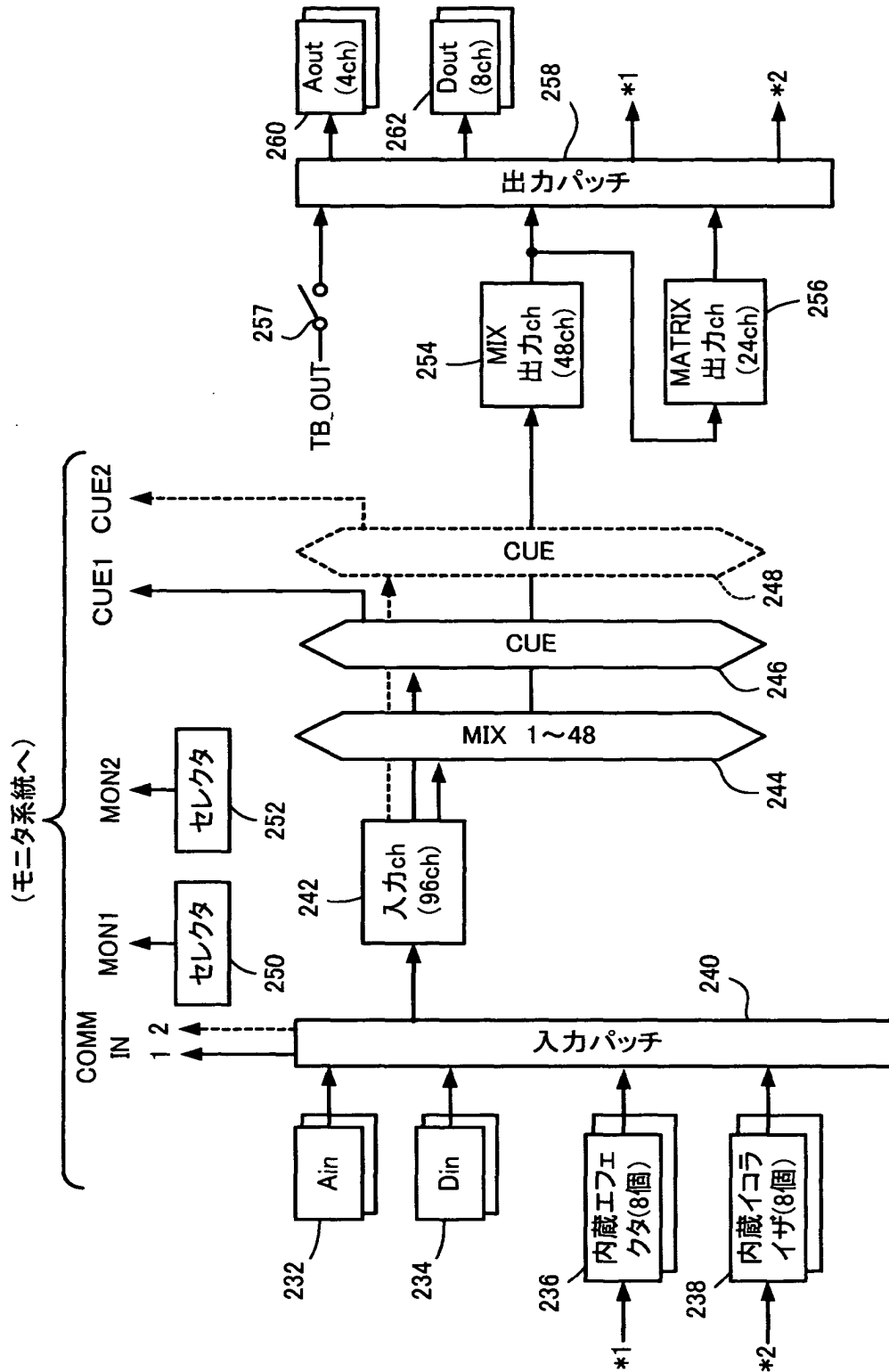
(d) デュアルコンソール+カスケード接続



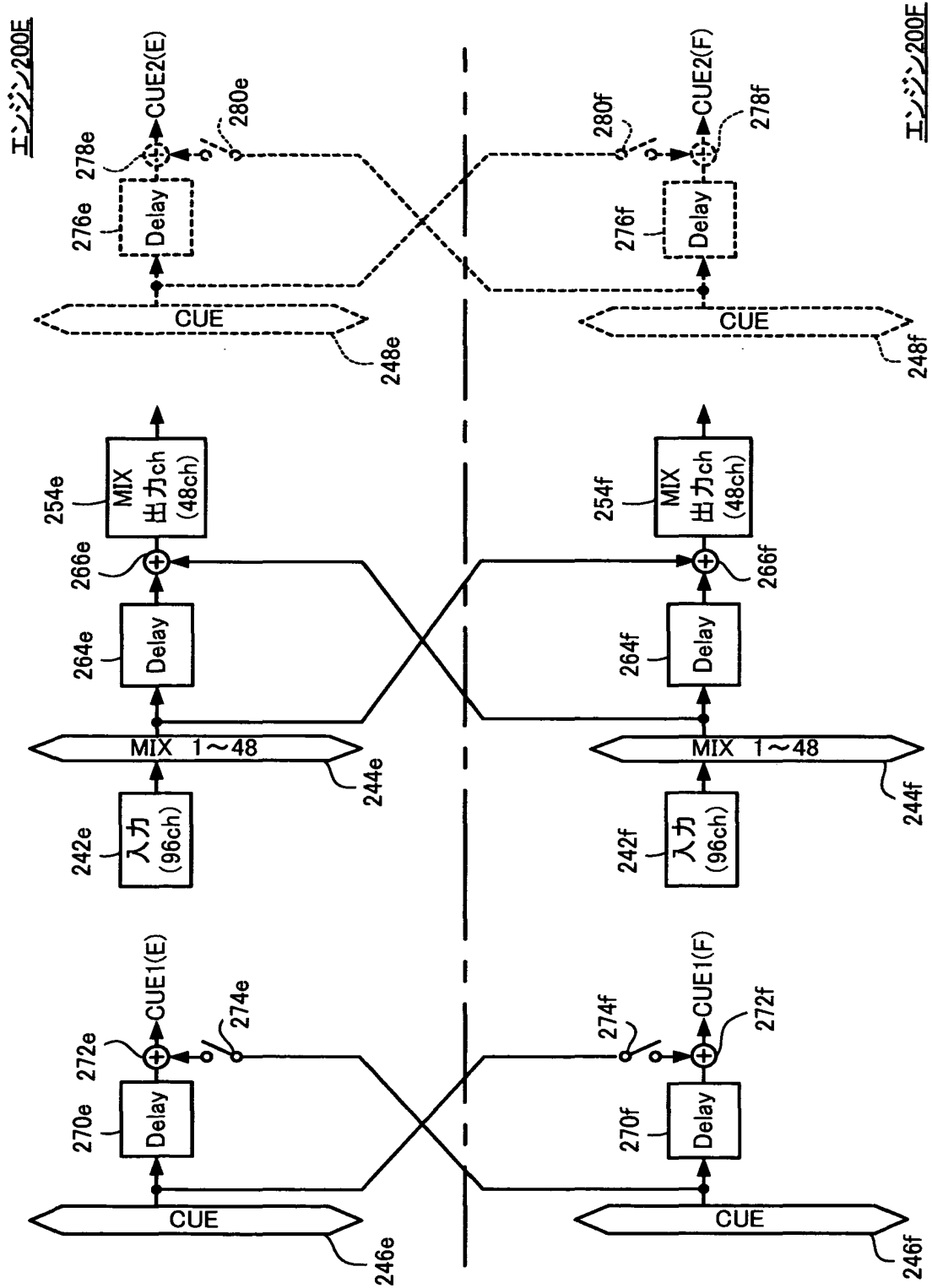
【図 3】



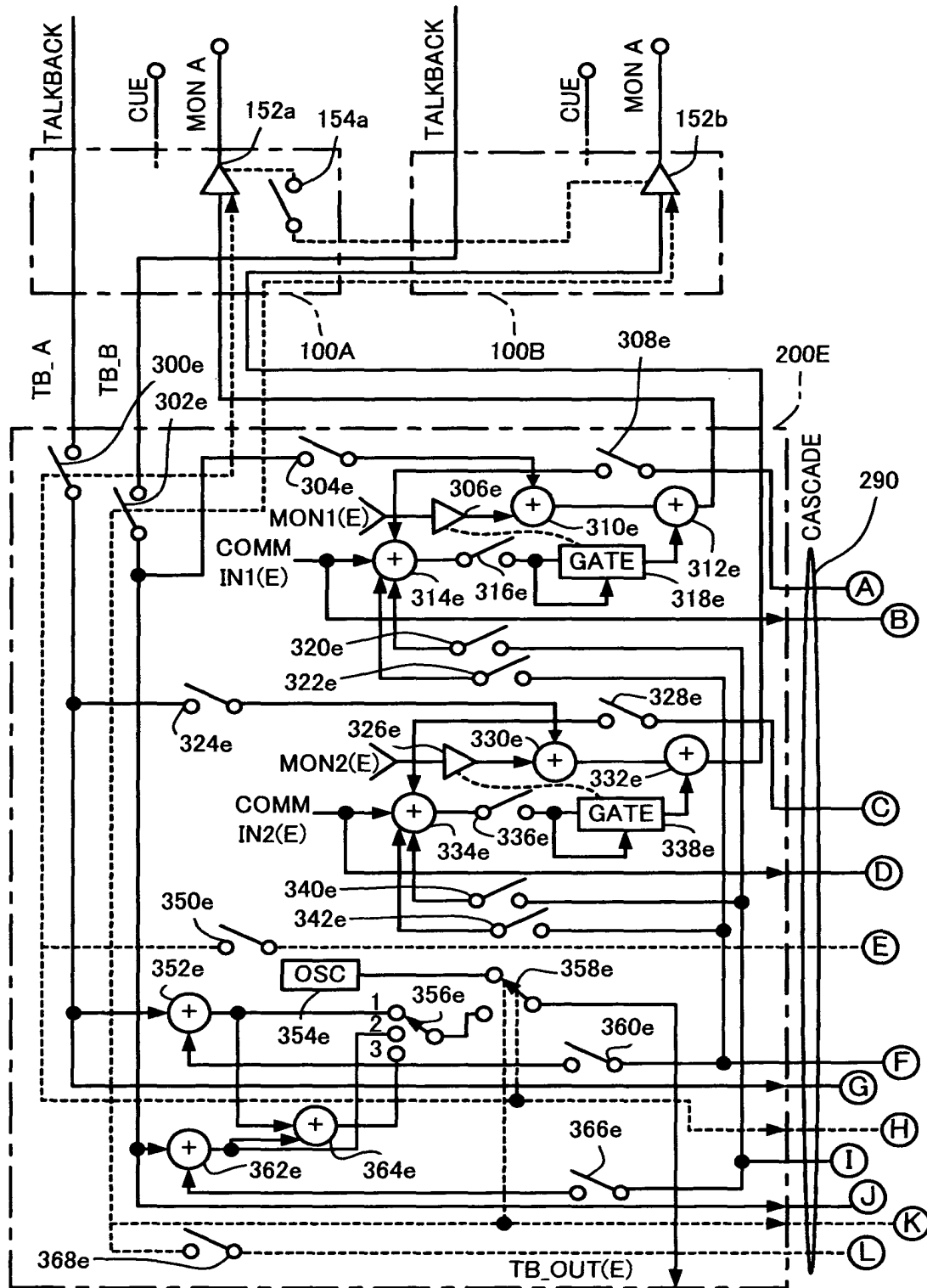
【図 4】



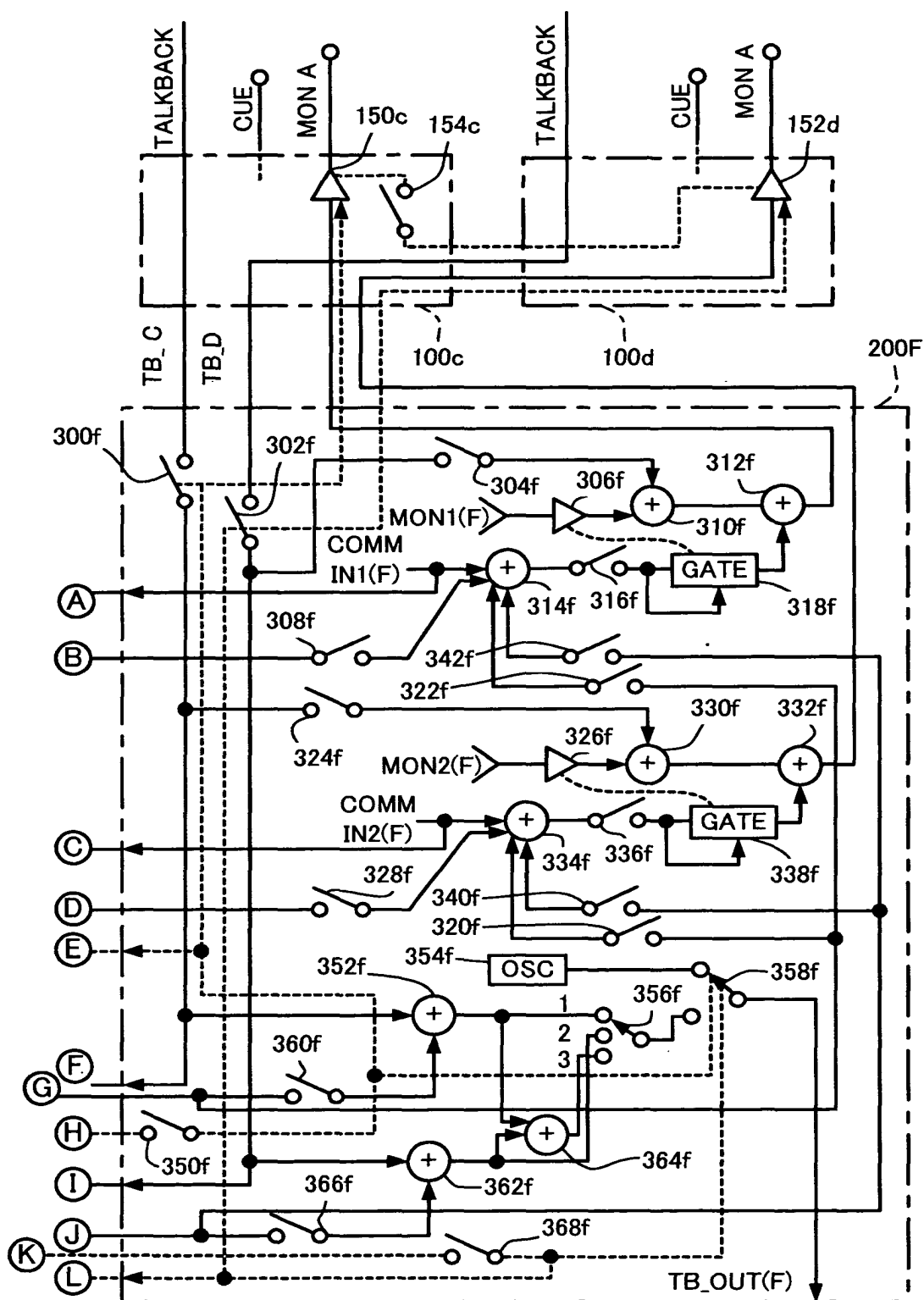
【図 5】



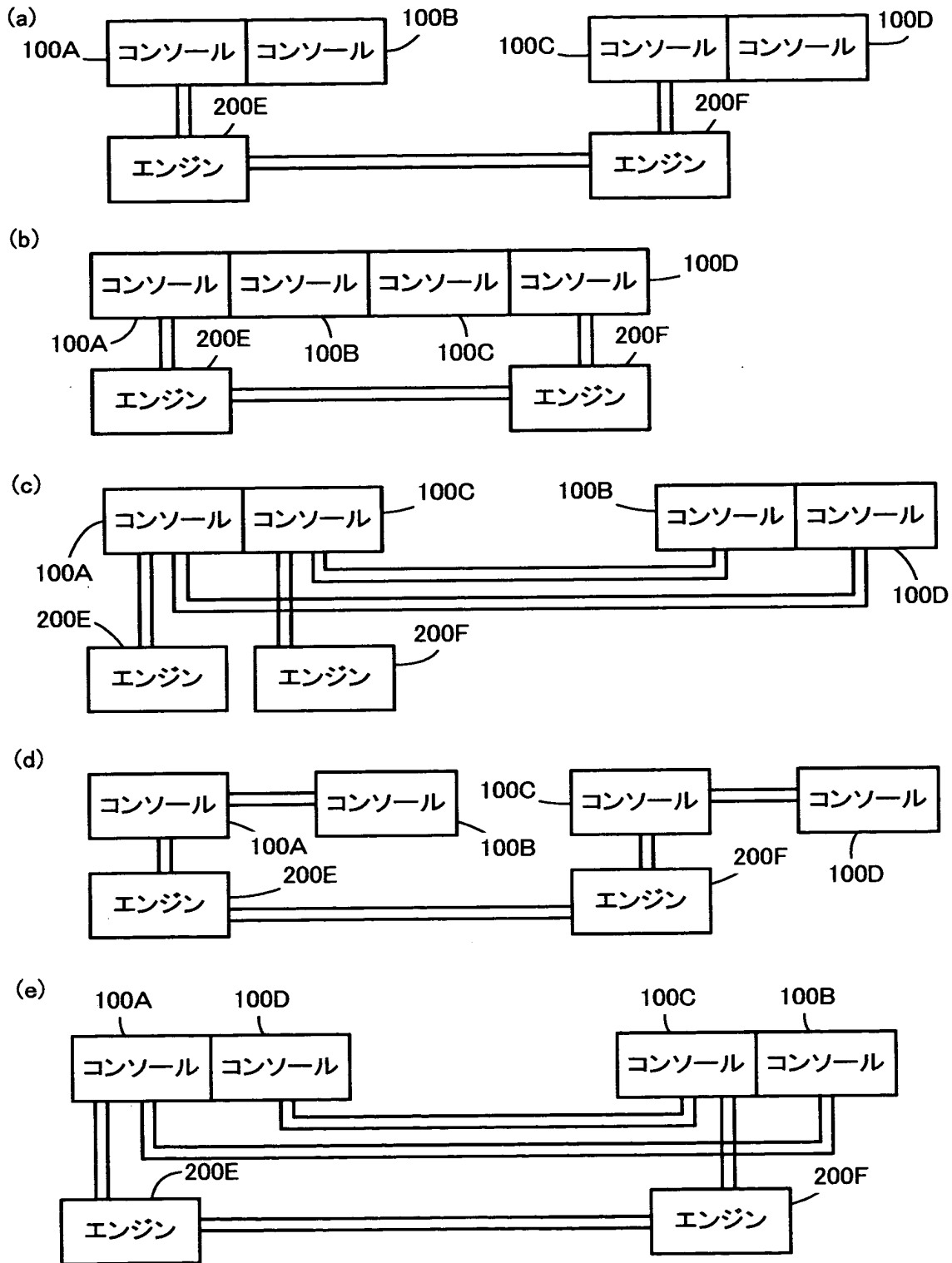
【図 6】



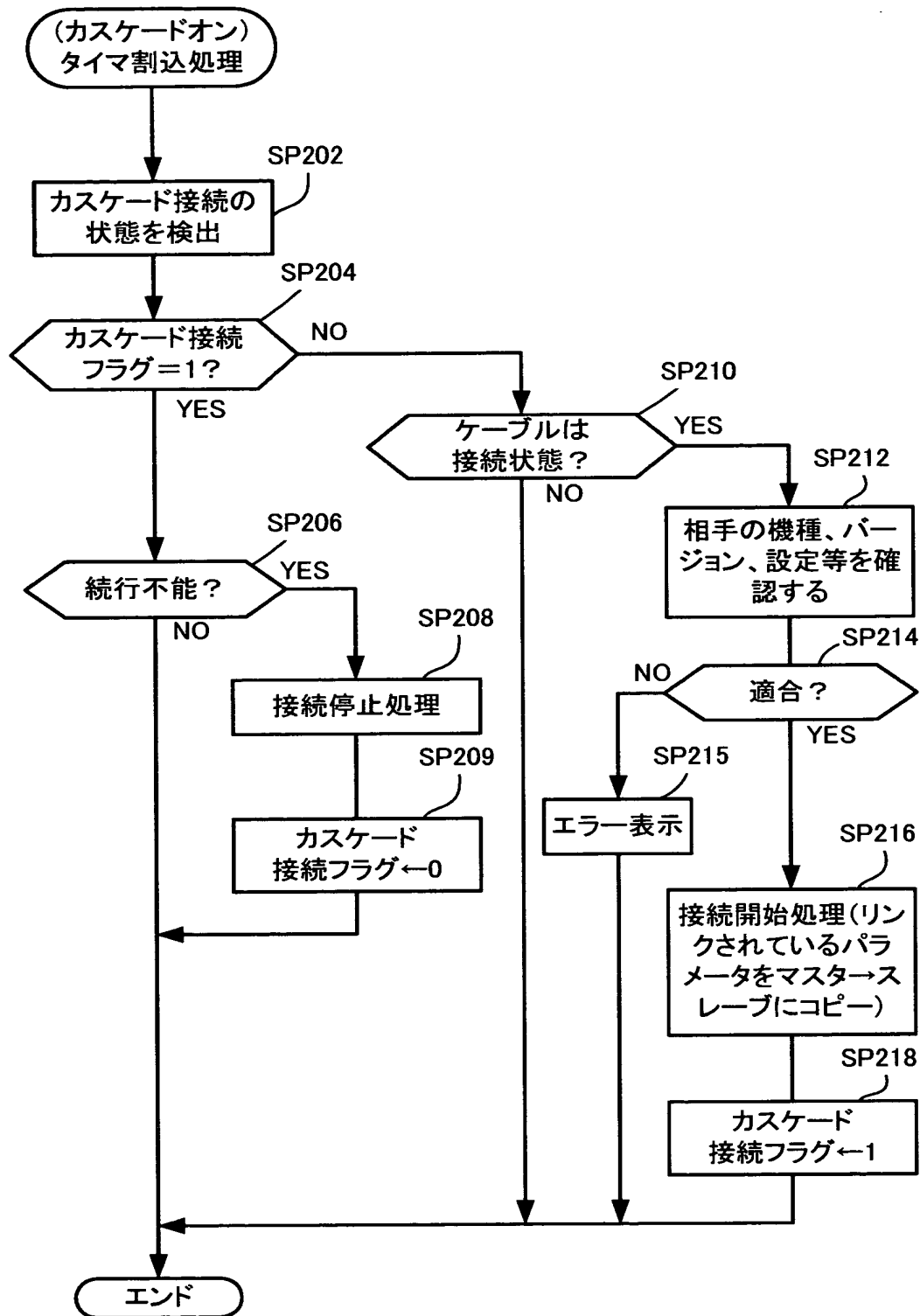
【図 7】



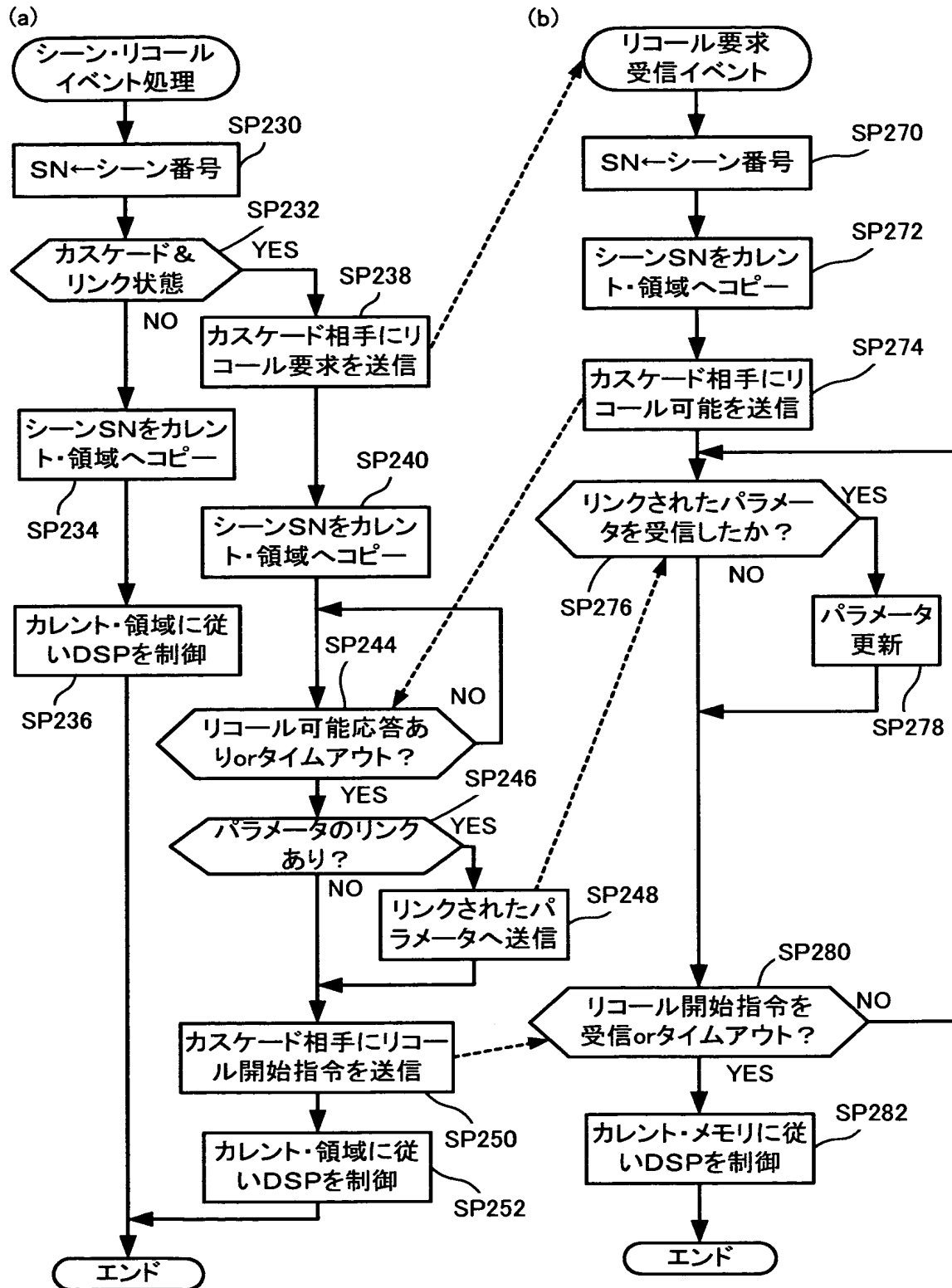
【図 8】



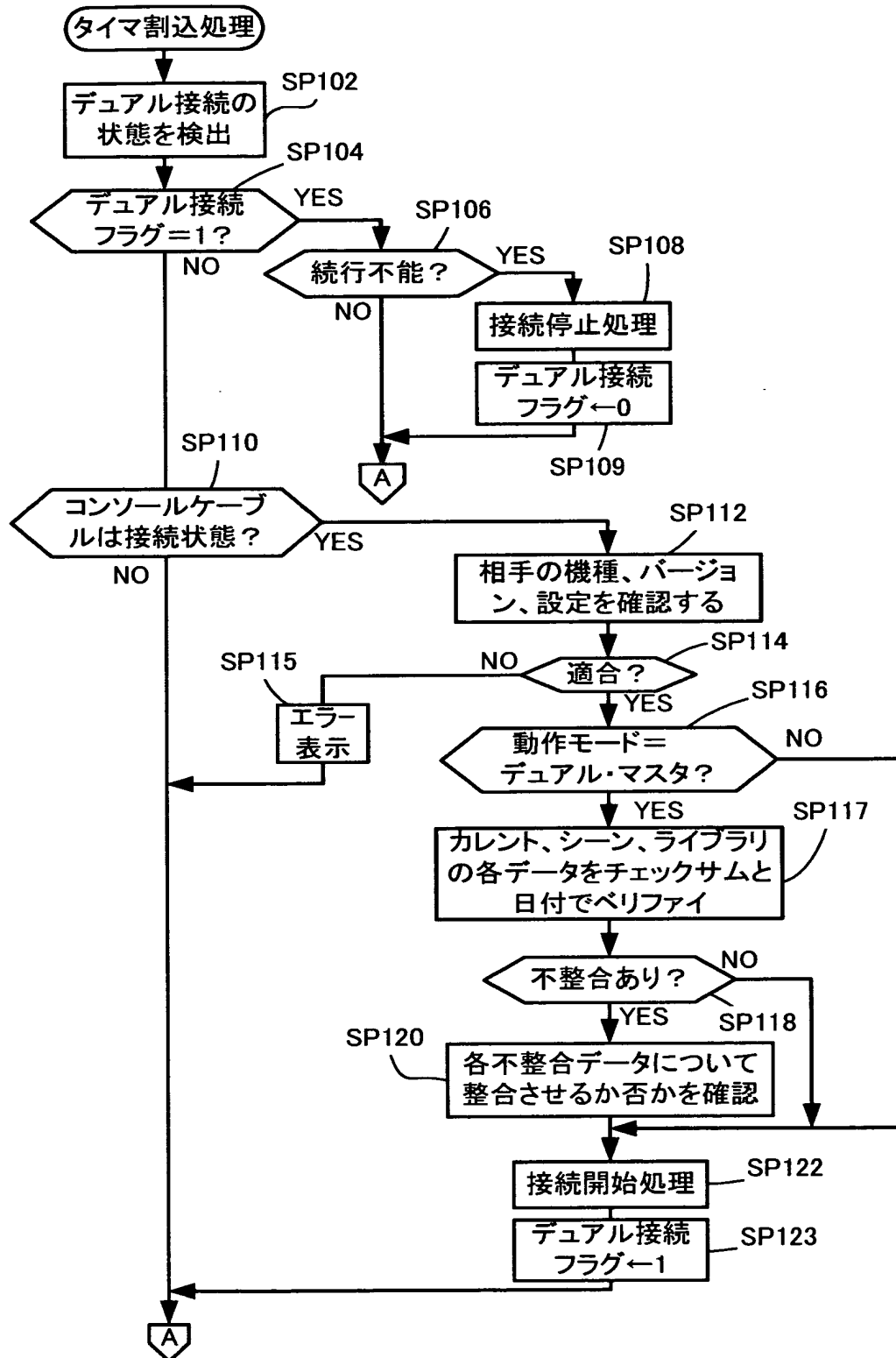
【図 9】



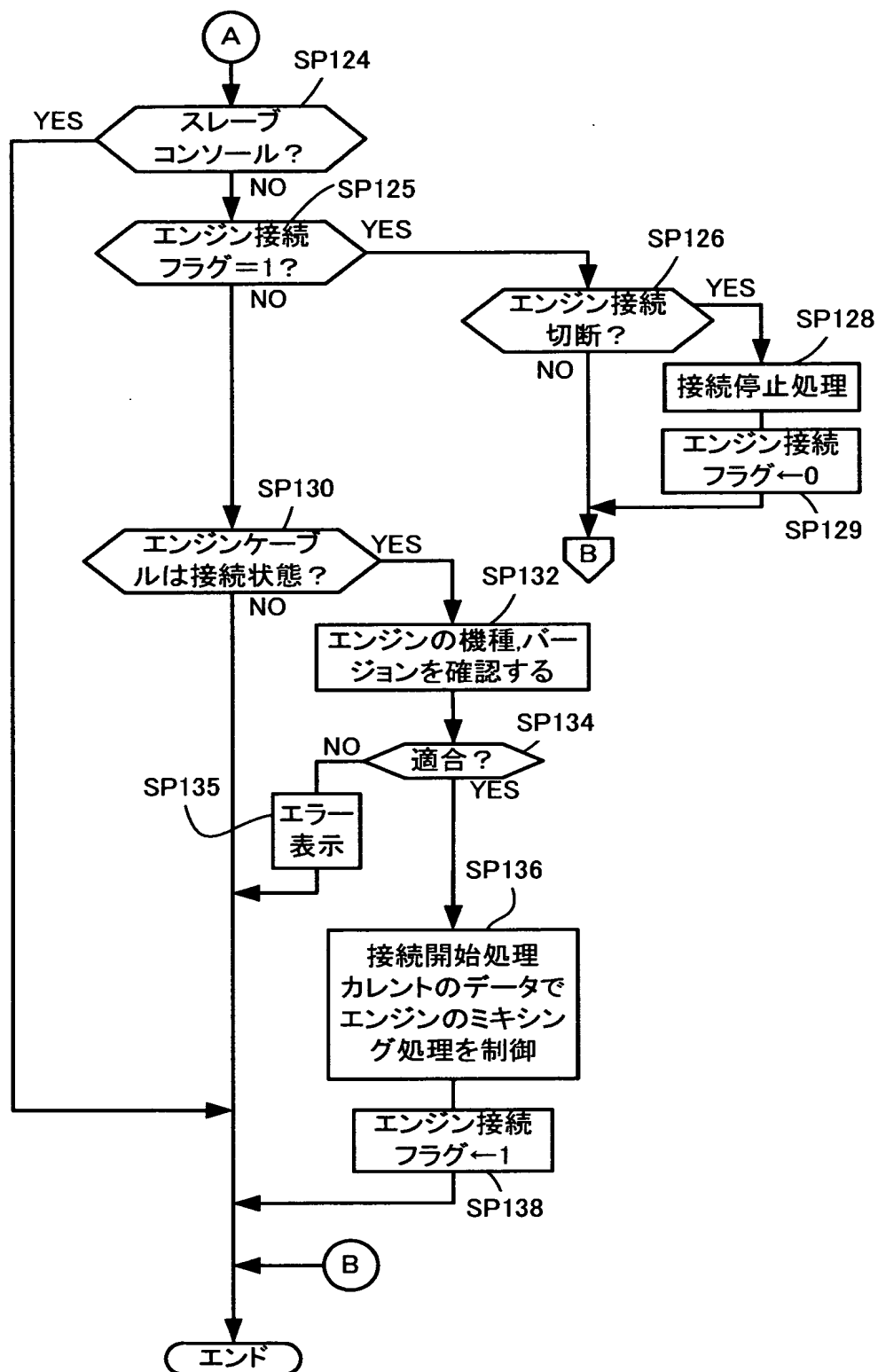
【図10】



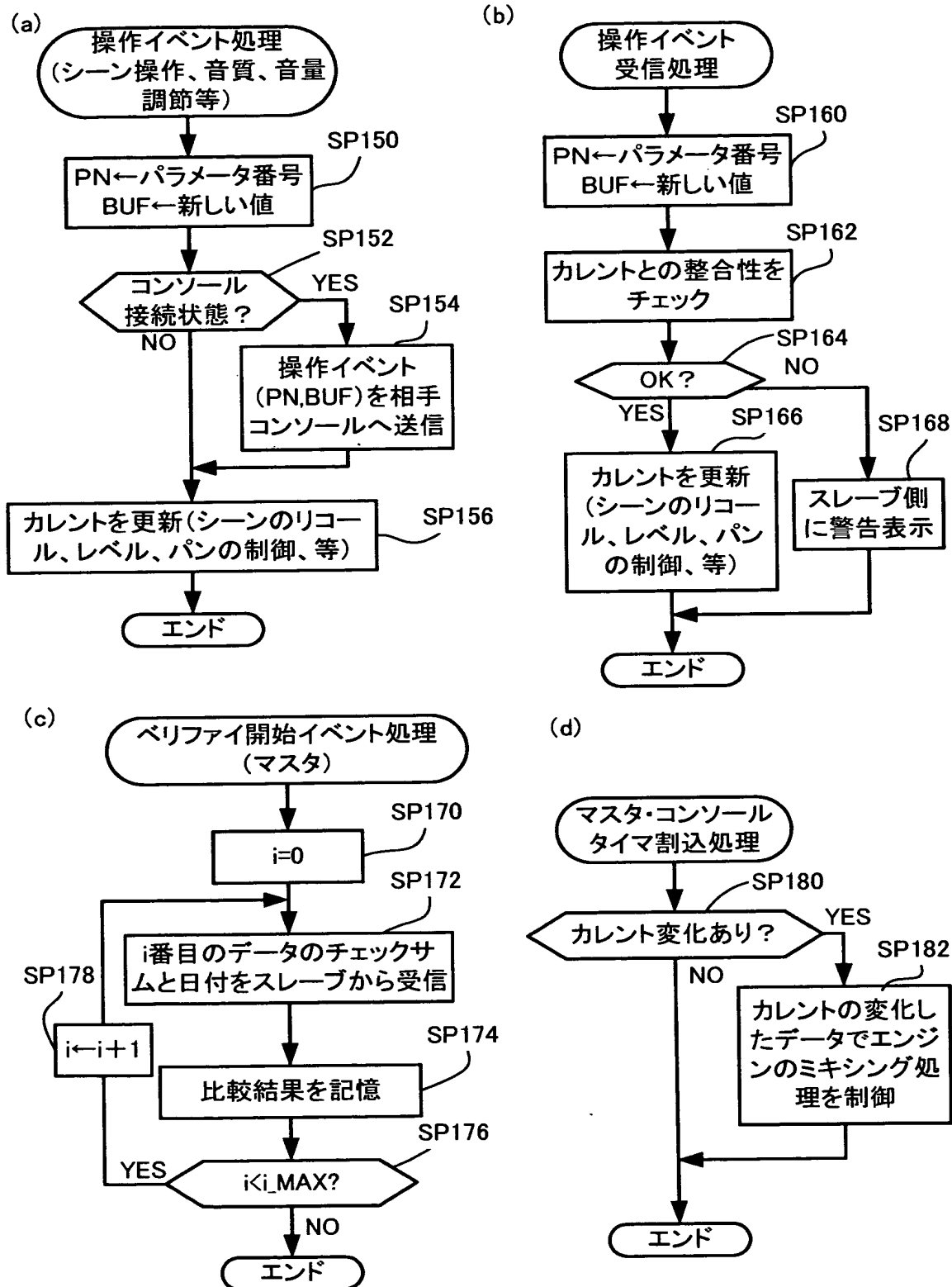
【図 1 1】



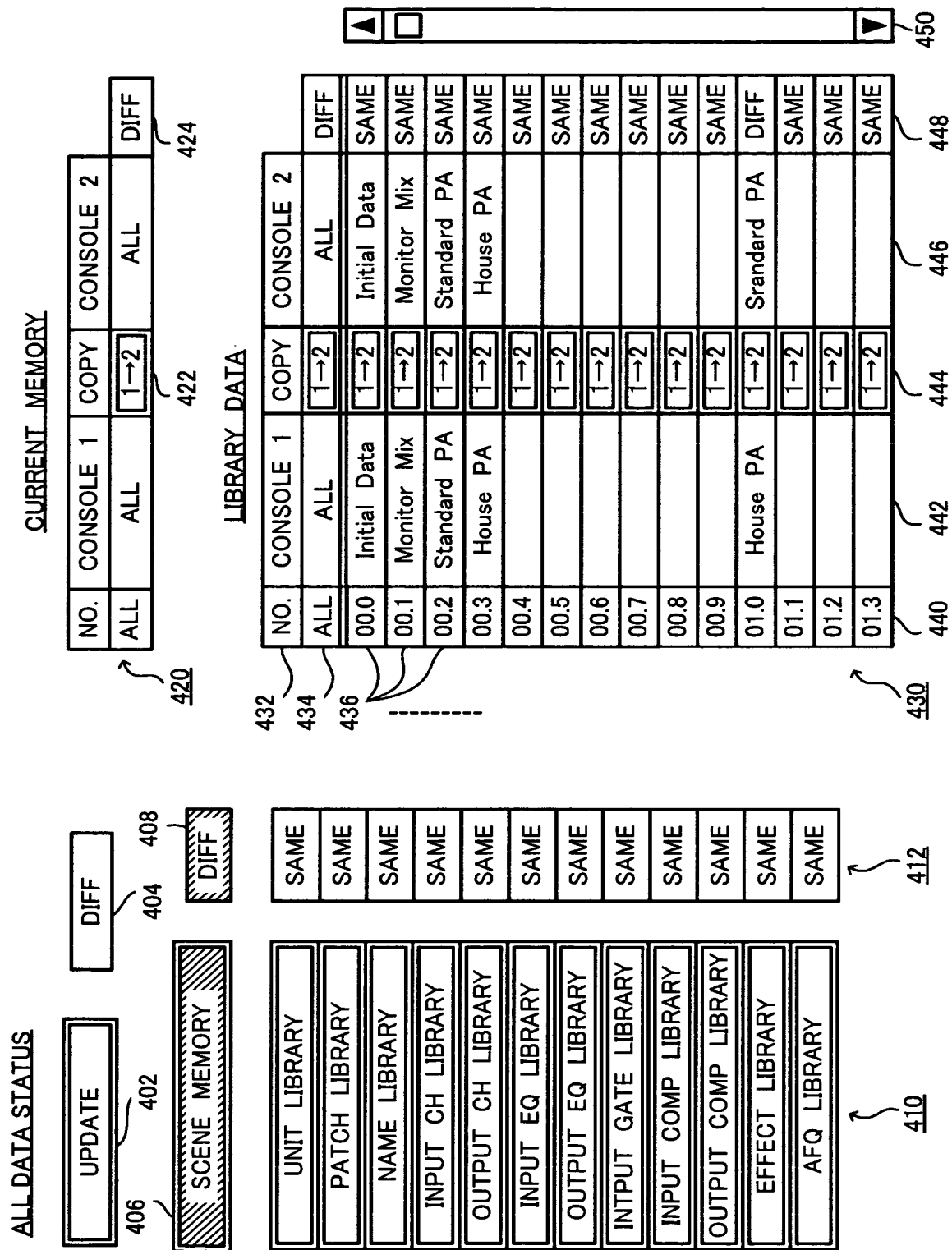
【図 12】



【図13】



【図 1 4】



【書類名】 要約書

【要約】

【課題】 カスケード接続された複数のミキシングシステムにおいて、各々独立して最終的なミキシング結果、または最終的なキュー信号のミキシング結果を得られるようにする。

【解決手段】 各ミキシングシステムにおいて、複数の入力信号を加算し入力加算信号を出力する第1の加算過程（ミキシングバス244e）と、該入力加算信号をカスケード信号として出力するカスケード出力過程（244eから加算器266fへの信号出力）と、他のデジタルミキサから出力されるカスケード信号を入力するためのカスケード入力過程（ミキシングバス244fから加算器266eへの信号入力）と、前記入力加算信号を遅延する遅延過程（遅延回路264e）と、遅延された前記入力加算信号と、入力されたカスケード信号とを加算しミキシング出力信号として出力する第2の加算過程（加算器266e）とを設けた。

【選択図】 図5

認 定 ・ 付 加 情 報

特許出願の番号	特願 2 0 0 2 - 2 2 0 9 4 3
受付番号	5 0 2 0 1 1 2 2 4 8 9
書類名	特許願
担当官	第八担当上席 0 0 9 7
作成日	平成 1 4 年 8 月 5 日

< 認定情報・付加情報 >

【提出日】	平成14年 7月30日
-------	-------------

出 願 人 履 歴 情 報

識別番号 [0 0 0 0 0 4 0 7 5]

1. 変更年月日	1 9 9 0 年 8 月 2 2 日
[変更理由]	新規登録
住 所	静岡県浜松市中沢町 1 0 番 1 号
氏 名	ヤマハ株式会社